

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2003-059196

(43)Date of publication of application : 28.02.2003

(51)Int.Cl.

G11B 20/12
G11B 7/0045
G11B 20/10
G11B 27/00
G11B 27/034
H04N 5/91

(21)Application number : 2002-005706

(71)Applicant : SHARP CORP

(22)Date of filing : 15.01.2002

(72)Inventor : KIYAMA JIRO
IWANO HIROTOSHI
YAMAGUCHI TAKAYOSHI

(30)Priority

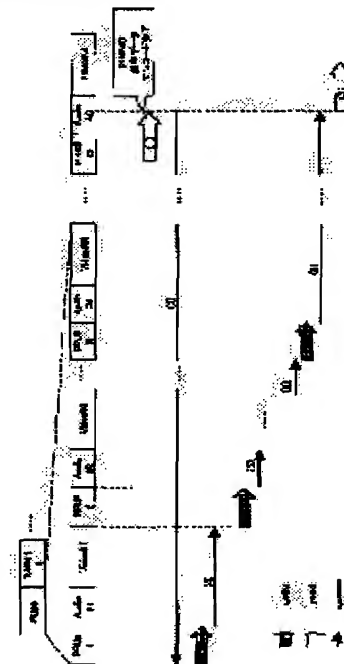
Priority number : 2001005826 Priority date : 15.01.2001 Priority country : JP
2001170444 06.06.2001 JP

(54) DATA RECORDING METHOD, DATA RECORDING APPARATUS, AND RECORDING MEDIUM

(57)Abstract:

PROBLEM TO BE SOLVED: To attain after-recording while reproducing an AV stream without interruption even when a disk drive with a comparatively low data transfer speed records the AV stream distributed on a disk.

SOLUTION: The data recording method is configured such that first data comprising video or audio data and second data reproduced synchronously with the first data are consecutively located on a recording medium to configure a first unit to record the data on the recording medium, and in the method the size of the recording unit of the first unit is decided on the basis of any of pickup mobile performance, a data transfer rate, a data bit rate, and second data rewrite control while reproducing the first data.



*** NOTICES ***

JPO and INPIT are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1]The 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data, It is a data recording method which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on a recording medium, control of said 2nd data rewrite while reproducing pickup moving performance, a data transfer rate, the bit rate of data, and said 1st data for a size of a record unit of said 1st unit, and ** — a data recording method characterized by what it opts for based on inner one.

[Claim 2]Said data recording method according to claim 1, wherein a maximum at the time of determining a size of a record unit of said 1st unit is set up.

[Claim 3]Said data recording method according to claim 1, wherein a minimum at the time of determining a size of a record unit of said 1st unit is set up.

[Claim 4]The 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data, It is a data recording method which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on a recording medium, memory quantity used when rewriting said 2nd data — pickup moving performance, a data transfer rate, the bit rate of data, control of said 2nd data rewrite, and ** — a data recording method characterized by what it opts for based on inner one.

[Claim 5]Said data recording method according to any one of claims 1 to 4, wherein control of said 2nd data rewrite is control which rewrites said 1st whole unit.

[Claim 6]Said data recording method according to any one of claims 1 to 4, wherein control of said 2nd data rewrite is control which rewrites only said 2nd data.

[Claim 7]Said data recording method according to claim 6 once control of said 2nd data rewrite reads an error correction block of the start edge of said 2nd data, and a termination which contains either at least, wherein it is performed.

[Claim 8]Said data recording method according to claim 7, wherein reading of said error correction block is performed at the time of reading of said 1st data.

[Claim 9]Said data recording method according to any one of claims 1 to 8, wherein said 1st unit comprises the 2nd one or more independent refreshable unit.

[Claim 10]The 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data, It is a data recording method which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on a recording medium, In the 2nd independent refreshable unit, it is constituted by said 1st unit and to said 2nd unit. It is contained by the 3rd unit that stores said 2nd data, and a size of a record unit of said 1st unit, Pickup moving performance, a data transfer rate, the bit rate of data, control of said 2nd data rewrite while reproducing said 1st data, and ** — a data recording method which is determined based on inner one and characterized by control of said said 2nd data rewrite being control rewritten every 3rd one or more unit.

[Claim 11]The 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data, It is a data recording method which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on a recording medium, Said 1st unit comprises the 2nd independent refreshable unit, A data recording method which control of said 2nd data rewrite while reproducing said 1st data is the control which rewrites only said 2nd data, and is characterized by determining a size of a record unit of said 1st unit based on regeneration time of said 2nd unit.

[Claim 12]Only the 1st data that consists of an image or a sound the 2nd data reproduced synchronizing with/and said 1st data, A data recording method which is a data recording method which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on a recording medium, and is characterized by changing a deciding method of a record unit of said 1st unit by a case where it is not considered as a case where said 2nd data exists.

[Claim 13]Said data recording method according to any one of claims 1 to 12, wherein a record unit of said 1st unit is specified by regeneration time.

[Claim 14]The 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data, It is a data recording method which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on a recording medium, A data recording method setting up the bit rate of a standard for securing a field in the 1st [for storing said 2nd data] unit independently of the bit rate of said 1st data.

[Claim 15]Said data recording method according to claim 14 making the bit rate of a standard for securing a field in said 1st unit into the greatest bit rate of said 2nd data.

[Claim 16]Said data recording method according to claim 15 making the bit rate of a standard for securing a field in said 1st unit into the bit rate lower than the bit rate of a sound in said 1st data.

[Claim 17]Only the 1st data that consists of an image or a sound the 2nd data reproduced synchronizing with/and said 1st data, When it is a data recording method which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on a recording medium and said 2nd data does not exist, A data recording method characterized by the 1st unit comprising said 2nd unit independent when said 1st unit comprises two or more 2nd units that are the units continuously arranged on a recording medium and said 2nd data exists.

[Claim 18]The 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data, A data recording method which is a data recording method which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on the 1st recording medium, and is characterized by what is once recorded on a record section on the

2nd recording medium when recording said 2nd data, reproducing said 1st data.

[Claim 19] Said data recording method according to claim 18 characterized by moving to said 1st unit on said 1st recording medium from a record section on said 2nd recording medium after said 2nd data recording.

[Claim 20] Said data recording method according to claim 18 or 19 recording only said 2nd data unrecordable on said 1st recording medium at the time of reproduction of said 1st data on said 2nd recording medium.

[Claim 21] Said data recording method according to any one of claims 18 to 20, wherein said 2nd recording medium is the same recording medium as said 1st recording medium.

[Claim 22] Said data recording method according to claim 21, wherein a record section on said 2nd recording medium is a field on said 1st unit.

[Claim 23] Said data recording method according to any one of claims 18 to 20, wherein said 2nd recording medium is semiconductor memory.

[Claim 24] A data recorder comprising:

The 1st data that consists of an image or a sound.

Arrange continuously the 2nd data reproduced synchronizing with said 1st data on a recording medium, and the 1st unit is constituted, it is a data recorder recorded on a recording medium — regeneration time of said 1st unit — pickup moving performance, a data transfer rate, the bit rate of data, control of the 2nd data rewrite, and ** — a means to determine based on inner one.

[Claim 25] A data recorder comprising:

The 1st data that consists of an image or a sound.

Arrange continuously the 2nd data reproduced synchronizing with said 1st data on a recording medium, and the 1st unit is constituted, A means to be a data recorder recorded on the 1st recording medium, and to once record on a record section on the 2nd recording medium when recording said 2nd data, reproducing said 1st data.

[Claim 26] The 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data are the recording media recorded, and Data for predetermined regeneration time in said 1st data, Manage the 2nd data reproduced synchronizing with this data as the 1st unit, and regeneration time of said 1st unit, pickup moving performance, a data transfer rate, the bit rate of data, control of the 2nd data rewrite, and ** — a recording medium characterized by what it opts for based on inner one.

[Claim 27] Are the 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data a data recording method recorded on a recording medium, and Data for predetermined regeneration time in said 1st data, A data recording method managing the 2nd data reproduced synchronizing with this data as the 1st unit, and determining regeneration time of said 1st unit based on the number of physical break points on a recording medium in said 1st unit.

[Claim 28] Said data recording method according to claim 27 controlling only said 2nd data to be physically recorded continuously on a recording medium.

[Claim 29] Said data recording method according to claim 27 or 28, wherein the 1st data in said 1st unit comprises independently a set of the 2nd unit that is a refreshable unit.

[Claim 30] It is a data recording method which records the 1st data that consists of sounds on a recording medium, A data recording method deciding on continuous recording time on a recording medium of said 1st data based on a maximum bit rate of the 2nd data that consists of an image and a sound which may carry out synchronous reproduction to said 1st data.

[Claim 31] A data recorder which records the 1st data that consists of an image or a sound characterized by comprising the following, and the 2nd data reproduced synchronizing with said 1st data on a recording medium.

Data for predetermined regeneration time in said 1st data.

A means to manage the 2nd data reproduced synchronizing with this data as the 1st unit.

A means to determine regeneration time of said 1st unit based on the number of physical break points on a recording medium in said 1st unit.

[Claim 32] The 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data are the recording media recorded, and Data for predetermined regeneration time in said 1st data, A recording medium, wherein it manages the 2nd data reproduced synchronizing with this data as the 1st unit and regeneration time of said 1st unit is based on the number of physical break points on a recording medium in said 1st unit.

[Translation done.]

* NOTICES *

JPO and INPIT are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention]This invention relates picture image data and voice data to record, the data recording method to play, and a data recorder to the recording medium in which random access, such as a hard disk and an optical disc, is possible.

[0002]

[Description of the Prior Art]The video and the audio digital recording playback equipment using a disk medium are spreading. In them, the art of realizing an after recording (postrecording) function cheaply like a tape medium is searched for. A postrecording function is a function which adds information, especially an audio afterwards to the audio and video which were already recorded.

[0003]As conventional technology which has realized the postrecording function using a disk medium, the disk recording playback equipment of the statement is known by JP,5-234084,A, for example.

[0004]After reading data into a memory from the disk which the reading period of data is playing from the program playing period now using a short thing before reading the following data, this art, It is possible to realize postrecording, even if it says that the inputted postrecording voice data is written in a disk and the number of disk recording reproduction means is one.

[0005]Here, a program playing period is between the peculiar regeneration phases which each of programs, such as video and music, have. For example, it cannot be said that the video for 1 minute was correctly played if it was not played in 1 minute even if the reproduction means changed.

[0006]The recording format of the disk in conventional technology is shown in drawing 22. A disk comprises a sequence of an ECC (error collection coding) block. An ECC block is the minimum unit at the time of coding, in addition to data, the parity for error amendment is added, and coding is performed.

[0007]Required data is taken out, after reading in this unit and carrying out an error correction, when reading data. On the other hand, when rewriting data, it reads per ECC block first, and to the data which carried out the error correction, a required portion is rewritten, error numerals are given again, and it records on a disk. This means that it is necessary to read and write in the whole ECC block in which the byte is contained, even when rewriting 1 byte.

[0008]In an ECC block, video and an audio are arranged in order of a postrecording audio block, an original audio block, and an original video block, as shown in drawing 22 (b).

[0009]The postrecording audio corresponding to the almost same time, an original audio, and original video are contained in each block. An original audio block and an original video block will be doubled, and it will be called an original block.

[0010]When recording an original program (image before recording a postrecording audio), dummy data is written in the postrecording audio block.

[0011]Next, the operation at the time of postrecording in conventional technology is explained with drawing 23. Here, the graph of drawing 23 (a) shows the time relation between reading from a disk, playback, or each processing called record. The sign in an arrow corresponds to the vertical axis in the graph of drawing 23 (b), and the position on the disk of the data used as a processing object is expressed.

Drawing 23 (b) shows typically the rate of program data that the graph of drawing 17 (c) occupies the position of the head in the inside of a disk to a buffer memory.

[0012]A program is arranged to the continuous field of s11 - s18- in a disk here, Each field of s11-s13, s13-s15, s15 - s17 assumes that it corresponds to an ECC block, respectively and each field of s11-s12, s13-s14, s15-s16, s17 - s18 supports the postrecording audio block, respectively.

[0013]It is at the time of the time t1, and the field to s13 is already stored in the buffer memory, and while the data currently recorded on the field of s11-s13 is decoded and reproduced, input of the postrecording sound of the data and encoding are performed.

[0014]In the time t1-t3, the data of the fields s13-s15 is read from a disk, and storing in a buffer memory and a postrecording buffer is performed. A postrecording buffer memorizes the read ECC block as it is, and takes the same composition as drawing 22 (b). The time t2 is time which decoding of the data currently recorded on the fields s11-s13 currently performed at the time of the time t1 and reproduction end.

[0015]While decoding the data of the fields s13-s15 read at the time t1-t3 after the time t2 and reproducing, input of the postrecording sound of the data and encoding are performed. Decoding of the data of these fields s13-s15 and reproduction are performed till the time t5.

[0016]Encoding will end at least the postrecording sound inputted by the time t2 by the time t3. In the time t3, the postrecording sound inputted by the time t2 is recorded on a disc medium. At this time, when accessing s11, the time of the rotational delay of a disk is required, but since it is a short time compared with the time of reading and writing of a disk, it does not take into consideration here.

[0017]The writing to the disk of a postrecording sound is performed at the time t3-t4. After the writing to this disk is completed at the time t4, the data of the time t4 to the fields s15-s17 is read from a disk. Thus, the same processing as the following is repeated.

[0018]At above-mentioned conventional technology, only one record reproduction means has realized postrecording by performing an information compression by using that read time becomes short, carrying out the time sharing of the record reproduction means, and using it by record and reproduction, rather than the regeneration time of data. The same art is

indicated by JP,2001-118362,A.

[0019]

[Problem(s) to be Solved by the Invention]However, in the conventional disk recording playback equipment mentioned above, when the bit rate of an AV stream is low enough compared with the data transfer rate of a disk, it is good, but when the margin of data input/output speed is small, it is difficult [it / postrecording] to break off and play.

[0020]In light of the above-mentioned problems, this invention is a comparatively low disk drive of a data transfer rate, and an object of this invention is to enable postrecording while breaking off and playing, even if the AV stream is moreover divided and recorded on the disk.

[0021]

[Means for Solving the Problem]An invention of the 1st of this application the 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data, It is a data recording method which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on a recording medium, control of said 2nd data rewrite while reproducing pickup moving performance, a data transfer rate, the bit rate of data, and said 1st data for a size of a record unit of said 1st unit, and ** — it determines based on inner one

[0022]A maximum at the time of an invention of the 2nd of this application determining a size of a record unit of said 1st unit is set up.

[0023]A minimum at the time of an invention of the 3rd of this application determining a size of a record unit of said 1st unit is set up.

[0024]An invention of the 4th of this application the 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data, It is a data recording method which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on a recording medium, memory quantity used when rewriting said 2nd data — pickup moving performance, a data transfer rate, the bit rate of data, control of said 2nd data rewrite, and ** — it determines based on inner one

[0025]An invention of the 5th of this application is characterized by control of said 2nd data rewrite being the control which rewrites said 1st whole unit.

[0026]An invention of the 6th of this application is characterized by control of said 2nd data rewrite being the control which rewrites only said 2nd data.

[0027]Control of said 2nd data rewrite is performed once an invention of the 7th of this application reads an error correction block of the start edge of said 2nd data, and a termination which contains either at least.

[0028]As for an invention of the 8th of this application, reading of said error correction block is performed at the time of reading of said 1st data.

[0029]The 9th invention of this application comprises the 2nd one or more unit with said 1st independently refreshable unit.

[0030]An invention of the 10th of this application the 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data, It is a data recording method which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on a recording medium. In the 2nd independent refreshable unit, it is constituted by said 1st unit and to said 2nd unit. It is contained by the 3rd unit that stores said 2nd data, and a size of a record unit of said 1st unit, Pickup moving performance, a data transfer rate, the bit rate of data, control of said 2nd data rewrite while reproducing said 1st data, and ** — it determines based on inner one and control of said said 2nd data rewrite is characterized by being the control rewritten every 3rd one or more unit.

[0031]An invention of the 11th of this application the 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data, It is a data recording method which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on a recording medium, Said 1st unit comprises the 2nd independent refreshable unit, Control of said 2nd data rewrite while reproducing said 1st data is the control which rewrites only said 2nd data, and a size of a record unit of said 1st unit is determined based on regeneration time of said 2nd unit.

[0032]An invention of the 12th of this application the 2nd data in which only the 1st data that consists of an image or a sound is reproduced synchronizing with/said 1st data, A deciding method of a record unit of said 1st unit is changed by a case where it is not considered as a case where it is a data recording method which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on a recording medium, and said 2nd data exists.

[0033]A record unit of said 1st unit is prescribed by regeneration time to the 13th invention of this application.

[0034]An invention of the 14th of this application the 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data, It is a data recording method which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on a recording medium. The bit rate of a standard for securing a field in the 1st [for storing said 2nd data] unit is set up independently of the bit rate of said 1st data.

[0035]An invention of the 15th of this application makes the bit rate of a standard for securing a field in said 1st unit the greatest bit rate of said 2nd data.

[0036]An invention of the 16th of this application makes the bit rate of a standard for securing a field in said 1st unit the bit rate lower than the bit rate of a sound in said 1st data.

[0037]An invention of the 17th of this application the 2nd data in which only the 1st data that consists of an image or a sound is reproduced synchronizing with/said 1st data, When it is a data recording method which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on a recording medium and said 2nd data does not exist, When said 1st unit comprises two or more 2nd units that are the units continuously arranged on a recording medium and said 2nd data exists, the 1st unit comprises said 2nd unit independent.

[0038]An invention of the 18th of this application the 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data, When recording said 2nd data, being a data recording method which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on the 1st recording medium, and reproducing said 1st data, it once records on a record section on the 2nd recording medium.

[0039]An invention of the 19th of this application moves to said 1st unit on said 1st recording medium from a record section on said 2nd recording medium after said 2nd data recording.

[0040]An invention of the 20th of this application records only said 2nd data unrecordable on said 1st recording medium at the time of reproduction of said 1st data on said 2nd recording medium.

[0041]An invention of the 21st of this application is characterized by said 2nd recording medium being the same recording

medium as said 1st recording medium.

[0042]An invention of the 22nd of this application is characterized by a record section on said 2nd recording medium being a field on said 1st unit.

[0043]An invention of the 23rd of this application is characterized by said 2nd recording medium being semiconductor memory.

[0044]An invention of the 24th of this application the 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data, It is a data recorder which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on a recording medium, regeneration time of said 1st unit — pickup moving performance, a data transfer rate, the bit rate of data, control of the 2nd data rewrite, and ** — it had a means to determine based on inner one

[0045]An invention of the 25th of this application the 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data, When recording said 2nd data, being a data recorder which arranges continuously on a recording medium, constitutes the 1st unit, and is recorded on the 1st recording medium, and reproducing said 1st data, it had a means to once record on a record section on the 2nd recording medium.

[0046]An invention of the 26th of this application is a recording medium recorded, and the 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data Data for predetermined regeneration time in said 1st data, managing the 2nd data reproduced synchronizing with this data as the 1st unit — regeneration time of said 1st unit — pickup moving performance, a data transfer rate, the bit rate of data, control of the 2nd data rewrite, and ** — it is determined based on inner one

[0047]An invention of the 27th of this application is the 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data a data recording method recorded on a recording medium, and Data for predetermined regeneration time in said 1st data, The 2nd data reproduced synchronizing with this data is managed as the 1st unit, and regeneration time of said 1st unit is determined based on the number of physical break points on a recording medium in said 1st unit.

[0048]An invention of the 28th of this application controls only said 2nd data to be physically recorded continuously on a recording medium.

[0049]An invention of the 29th of this application comprises a set of the 2nd unit whose 1st data in said 1st unit is a refreshable unit independently.

[0050]An invention of the 30th of this application is a data recording method which records the 1st data that consists of sounds on a recording medium, Based on a maximum bit rate of the 2nd data that consists of an image and a sound which may carry out synchronous reproduction to said 1st data, it decides on continuous recording time on a recording medium of said 1st data.

[0051]Data recorder of this invention recorded on a recording medium is characterized by that the 31st invention of this application comprises the following in the 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data.

Data for predetermined regeneration time in said 1st data.

A means to manage the 2nd data reproduced synchronizing with this data as the 1st unit.

A means to determine regeneration time of said 1st unit based on the number of physical break points on a recording medium in said 1st unit.

[0052]An invention of the 32nd of this application is a recording medium recorded, and the 1st data that consists of an image or a sound, and the 2nd data reproduced synchronizing with said 1st data Data for predetermined regeneration time in said 1st data, The 2nd data reproduced synchronizing with this data is managed as the 1st unit, and regeneration time of said 1st unit is based on the number of physical break points on a recording medium in said 1st unit.

[0053]

[Embodiment of the Invention]Hereafter, the embodiment of this invention is described in detail, referring to drawings.

[0054]<System configuration> drawing 1 is a block diagram which is used in common in this embodiment and in which showing the outline composition of the video disc recorder which can be postrecorded. As shown in drawing 1, this device, The bus 100, the host CPU 101, RAM102, ROM103, the user interface 104, the system clock 105, the optical disc 106, the pickup 107, ECC decoders 108, ECC encoder 109, the buffer 110 for playback, The buffer 111 for record/postrecording, the appearance multiplexer 112, the multiplexer 113, the buffer 114 for multiplexing, the audio decoder 115, the video decoder 116, the audio encoder 117, the video encoder 118 and the camera that is not illustrated, a microphone, a loudspeaker, It has the display etc.

[0055]Although the host CPU 101 has not carried out the demultiplexer 112, the multiplexer 113, the pickup 107, and the graphic display through the bus 100, it performs communication with the audio decoder 115, the video decoder 116, the audio encoder 117, and the video encoder 118.

[0056]At the time of playback, the error correction of the data read from the optical disc 106 through the pickup 107 is carried out by ECC decoders 108, and it is once stored in the buffer 110 for playback. The demultiplexer 112 distributes the data in the buffer for reproduction to a suitable decoder by the classification according to the data transmission request from the audio decoder 115 and the video decoder 116.

[0057]On the other hand, at the time of record, it is once sent to the buffer 114 for multiplexing, and AV multiplexing is carried out by the multiplexer 113 and the data by which compression encoding was carried out with the audio encoder 117 and the video encoder 118 is sent to the buffer 111 for record/postrecording. By ECC encoder 109, an error correcting code is added and the data in the buffer 111 for record/postrecording is recorded on the optical disc 106 through the pickup 107.

[0058]MPEG-1 Layer-II is used for the coding mode of audio information, and MPEG-2 is used for the coding mode of a video data, respectively.

[0059]Let the optical disc 106 be an optical disc in which record reproduction is spirally performed toward inner circumference from a periphery and which can be desorbed. 2048 bytes is used as one sector and an ECC block consists of 16 sectors for an error correction. When rewriting the data in an ECC block, after reading the whole ECC block in which the data is contained, performing an error correction and rewriting the target data, it is necessary to add an error correcting code again, to constitute an ECC block, and to record on a recording medium.

[0060]In order that the optical disc 106 here may raise recording efficiency, ZCAV (zone constant angular velocity) is adopted, and a record section comprises several zones where number of rotations differs.

[0061]In a <file system>, next this embodiment, in order to manage the variety of information on the optical disc 106, a file

system is used. In consideration of interoperability with PC, UDF (Universal Disk Format) is used for a file system. On a file system, various management information and an AV stream are treated as a file.

[0062]User area is managed by 2048 bytes of logical block (a sector and a one to one correspondence). There is no necessity that each file comprises a logical block which continued physically on the disk, and it distributes per logical block and may be recorded. Free space is managed per logical block using Space Bitmap.

[0063]<File format> By this embodiment, a QuickTime file format is used as a format for AV stream management again. A QuickTime file format is a format for multimedia data management which Apple developed, and it is used widely in the world of the personal computer (PC).

[0064]A QuickTime file format comprises management information and an AV stream. Here, both are doubled and it is called a QuickTime movie. Both may exist in the same file or may exist in a separate file. When it exists in the same file, composition as shown in drawing 2 (a) is taken. A variety of information is stored in a common structure of atom. Management information is stored in the structure of Movie atom, and an AV stream is stored in the structure of Movie data atom.

[0065]The attribution information of the table for drawing the relative position of a under [the file of the AV information corresponding to the arbitrary time in an AV stream], audio information, or a video data, the external reference information mentioned later, etc. are included in the management information in Movie atom.

[0066]On the other hand, when management information and an AV stream are stored in a separate file, composition as shown in drawing 2 (b) is taken. An AV stream does not need to be stored in atom although management information is stored in the structure of Movie atom. At this time, it is said that Movie atom is carrying out "external reference" of the file which stored the AV stream.

[0067]External reference can be carried out to two or more AV stream files, as shown in drawing 2 (c), and what is called "non-linear editing" and "non-destroying edit" which are shown as edited seemingly are attained according to this structure, without moving the AV stream itself physically.

[0068]The composition of Movie atom is shown in drawing 3 (a). Each atom has composition which includes specific atom. Movie atom contains one or more Track atom etc. which store the information about each track included in Movie header atom which manages the overall attribute of the program which the Movie atom manages, or its program.

[0069]Although atom which stores various kinds of information is included in each atom, it extracts and explains to what is needed for an understanding of this invention here. atom for managing the original information which is not defined by QuickTime format called User data atom is storable in Movie atom.

[0070]In this invention, in User data atom, as shown in drawing 3 (b), record-unit description atom which stores the information about the structure (composition of Record Unit mentioned later or Video Unit) of an AV stream, The additional definition of X descriptor atom which manages set performance atom which stores the information about the performance of apparatus required to reproduce the program, including seek time, a disc transfer rate, etc., is carried out.

[0071]The <1st example>, next the 1st example in this invention are described with drawing 4 thru/or drawing 11.

[0072]<the gestalt of an AV stream> — the composition of the AV stream in this example is first explained using drawing 6 from drawing 4. As shown in drawing 4, an AV stream comprises Record Unit (RU) of integer pieces. RU is a unit continuously recorded on a disk.

[0073]The length of RU, However it may arrange on a disk RU which constitutes an AV stream, it will be set up so that seamless playback (it can play without a picture and a sound breaking off during playback), and real-time postrecording (record an audio, carrying out seamless playback of the video for postrecording) may be guaranteed.

[0074]This setting method is mentioned later. A stream is constituted so that RU boundary may be in agreement with an ECC block boundary. Arrangement of RU unit can be easily changed on a disk, guaranteeing seamless playback with these character of RU, even after recording an AV stream on a disk.

[0075]RU consists of Video Unit (VU) of integer pieces. VU is an independent refreshable unit and can serve as an entry point in the case of reproduction from that. The composition of VU differs by a not corresponding stream (stream corresponding to un-[postrecording]) in the stream (stream corresponding to postrecording) and postrecording corresponding to postrecording.

[0076]First, VU composition in the stream corresponding to un-[postrecording] is shown in drawing 5. VU comprises AAU (audio access unit) of the integer pieces which stored the main audio information reproduced at the same time as GOP (group OBU picture) of integer pieces and them which stored the video data for about 1 second. GOP is a unit of the compression in MPEG-2 video standard, and comprises two or more video frames (typically about 15 frames).

[0077]AAU is a unit of the compression in an MPEG-1 LayerII standard, and is constituted by the sound-wave-forms sample point of 1152 points. When a sampling frequency is 48 kHz, the regeneration time per AAU will be 0.024 second. In VU, in order to make small delay which is needed for AV synchronous reproduction, it arranges in order of AAUGOP.

[0078]The reproduction start timing of the head video frame in VU must be before the reproduction start timing of the head AAU, and the time lag must be less than the regeneration time of 1AAU. VU is beyond a predetermined unit and it is necessary to record it continuously on a disk for seamless playback. The unit is mentioned later.

[0079]VU unit — independence — SequenceHeader (SH) is placed at the head of the video data in VU so that it may be refreshable. When the attribute (for example, the number of pixels which constitutes a screen) of VU which follows, and video encoding changes, Sequence End Code (SEC) is placed at an end.

[0080]The regeneration time of VU is defined as having applied the video frame cycle to the number of video frames contained in VU. The end of VU is filled up with 0 in order to double the always end of RU with an ECC block boundary, when it constitutes RU combining several ready VU.

[0081]On the other hand, the composition of VU in the stream corresponding to postrecording is as being shown in drawing 6. Post Recording Unit (PRU) is provided as a field for storing the postrecording (subaudio) data reproduced simultaneously with video and main audio information at the head of VU in the stream corresponding to un-[postrecording]. Reverse may be sufficient although PRU is placed in front of the field for storing a main audio here.

[0082]The area size of PRU is not concerned with the bit rate of a main audio, but is chosen from one kind or the limited kind. Because, when setting out of the bit rate is enabled freely, it is because the apparatus with a postrecording function needs to support audio encoding by all the bit rates.

[0083]For example, the area size of PRU is not concerned with the bit rate of a main audio, but is secured based on the greatest refreshable audio bit rate. For example, even if the bit rate of a main audio is 128k bps, the area size of PRU contained in the same VU is secured with a refreshable maximum bit rate (for example, 256k bps).

[0084]In this case, when postrecording by another apparatus, it is not bound by the bit rate of the audio of the apparatus which recorded the AV stream from the first, but a postrecording audio can be recorded by the bit rate which that apparatus supported, and the burden corresponding to encoding in the apparatus which postrecords decreases.

[0085]There is also a method of not being concerned with the bit rate of a main audio, but on the other hand, securing the area size of PRU based on the low audio bit rate. Although not usually postrecorded by this, it also enables ** and others to save the storage capacity of a disk, if small for many users that necessity occurs very occasionally. Since it is human being's sound in many cases, a postrecording input is satisfactory in tone quality, when the most, even if it is the low bit rate.

[0086]Although the data which the object of postrecording records in video, and a main audio and postrecording is made into subaudio information here, the following explanation in particular is not limited to it.

[0087>About the position and regeneration time of each RU on a <AV stream controlling method> AV stream, or each VU, it manages by above-mentioned Movie Atom. For details, since it is unnecessary to explanation here, it omits.

[0088]<Disk arrangement deciding method> The deciding method of RU regeneration time in the stream corresponding to postrecording is explained first. The postrecording algorithm (reference postrecording algorithm) which serves as a device (reference device model) used as a standard and a standard in this deciding method for the compatibility reservation between apparatus is assumed. Next, when it postrecords using them, RU regeneration time is decided that seamless reproduction does not fail.

[0089]Then, a reference device model is first explained using drawing 7. The ECC encoder decoder 501, the track buffer 502, the demultiplexer 503 and the buffer 504 for postrecording with which a reference device model is connected with one pickup at it, the audio encoder 509, the video buffer 505, It is constituted by the audio buffer 506, the video decoder 507, and the audio decoder 508.

[0090]In this model, since a pickup is one piece, time sharing performs record on the disk of read-out from the disk of the data for playback, and the data for postrecording. When reading the data for playback from a disk, it reads also including PRU. The ECC block (PRU block) containing read PRU is sent to the buffer 504 for postrecording from the track buffer 502.

[0091]The audio encoder 509 is outputted to the buffer 504 for postrecording an AAU cycle. With this output, the corresponding PRU block in the buffer 504 for postrecording is overwritten. Record of postrecording data is performed by recording a PRU block on a predetermined ECC block.

[0092]In this model, assuming sending a PRU block to the buffer 504 for postrecording from the track buffer 502 is based on the following reason. In the AV stream in this example, since a PRU boundary and an ECC block boundary are not in agreement, the data (the video data of last VU and audio information of the same VU) of not only the data of PRU but others is contained in an ECC block including a PRU boundary.

[0093]Therefore, when recording data on PRU, it is necessary to once read an ECC block including a PRU boundary to a memory. Although reading to a memory just before recording PRU is also considered, Since the ECC block which includes a PRU boundary at the time of the data read for reproduction is always read, the ECC block containing PRU read at the time of the data read for reproduction by holding to the buffer 504 for postrecording temporarily. an ECC block including a PRU boundary — read-out is omitted again.

[0094]The seamless reproduction in this model shall be guaranteed if at least one-piece VU exists on the track buffer 502 at the time of the decoding start of VU. Output speed of data is set to Rs from the entry-of-data speed and ECC decoders 501 to ECC encoder 501 of audio frame data.

[0095]The maximum term by access which is read and record stops is set to Ta. Time which short access (about 100 tracks) takes is set to Tk. Seek time, latency speed, and time until the data first read from the disk after access is outputted from ECC are contained in these periods. In this example, it may be Tk= 0.2 second for Rs=20Mbps, and Ta= 1 second.

[0096]Next, a reference postrecording algorithm is explained using drawing 8. The numbers from (1) to (9) in drawing 8 are equivalent to the numbers from (1) to (9) under following explanation. The outline of an algorithm is as follows.

[0097](1) Read the data for reproduction. (2) Perform pickup movement to RU#N at the same time encoding of the audio information corresponding to RU#N which is Nth RU is completed. (3) Record the PRU block corresponding to PRU#1 which is PRU of the head of RU#N.

[0098](4) Move a pickup to PRU#2 which is 2nd PRU in RU#N. (5) Record the PRU block corresponding to PRU#2. (6) Repeat pickup movement to the next PRU, and PRU block record. (7) Record the PRU block corresponding to PRU#M which is PRU of the last in RU#N. (8) Return a pickup to the original reading position. (9) Resume read-out of the data for reproduction. The above operation is repeated.

[0099]In said reference device model, if the following conditions are fulfilled when it postrecords using said reference postrecording algorithm, it can guarantee that there is no underflow of the track buffer 502.

[0100]When maximum recording time of PRU in Tr(i) and RU#i is made into Tw(i) for the maximum reading time which includes T (i) and a division jump for the maximum regeneration time about RU#i in which the condition is arbitrary RUs in an AV stream, Te(i)>=Tr(i)+Tw(i) ... < type It is that 1> is materialized.

[0101]Because, this formula can be set to arbitrary n which is a sufficient condition of seamless reproduction. [0102]

[Equation 1]

$$\sum_{i=1}^n Te(i) \geq \sum_{i=1}^n (Tr(i) + Tw(i)) \quad \cdot \cdot \cdot < \text{式 } 2 >$$

[0103]It is because it is a ***** sufficient condition.

[0104]Since record on the disk of postrecording data is performed synchronizing with the completion of PRU encoding, the data in the buffer 504 for postrecording does not accumulate, and there is also no overflow of the buffer 504 for postrecording.

[0105]< type When Tr(i) in 1> sets the maximum bit rate of the audio used as the standard of the audio in an AV stream, video, and PRU area-size reservation to Ra, Rv, and Rp, respectively, Tr(i)=Te(i)x(Rv+Ra+Rp)/Rs+Ta ... < type It becomes 3>.

[0106]The 1st paragraph of the right-hand side expresses the reading time of RU#i. The 2nd paragraph of the right-hand side expresses the maximum access time by the division jump generated immediately after RU#i read-out. Tw(i) — Tw(i)=2Ta+(M-1) xTk+Te(i)xRp/Rs+(2M-1) xLy/Rs ... < type It becomes 4>.

[0107]Here, the 1st paragraph of the right-hand side shows both-way access time to RU. Using maximum-access-time Ta for access time of a round trip of PRU is based on the following reasons.

[0108]It depends on a time delay by a buffer for reproduction at that time for distance of a track read now and a track with which PRU which should be recorded exists. However, a time delay changes with buffer size for reproduction, and even if it is the same buffer size, also when read-out stops temporarily by a shock immediately before, they differ. That is, distance to access is unfixed, therefore it is necessary to estimate it with the worst value.

[0109]The 2nd paragraph of the right-hand side is the sum total of time to jump between PRU. M is the number of VU which constitutes RU#. The 3rd paragraph of the right-hand side expresses the sum total of time for recording PRU contained in RU# on a disk. The 4th paragraph of the right-hand side expresses the maximum of the record time other than postrecording data in an ECC block in which PRU both ends are included.

[0110]Here, L_y is set to 32 KB which is ECC block size. Since not both ends of PRU are necessarily in agreement with an ECC block boundary, a reason which needs such a paragraph is for recording mostly by a maximum of 2 ECC blocks from size of PRU at the time of PRU record. However, since PRU of a head of RU is located in an ECC block boundary, it is $(2M-1)$.

[0111]Here, in order that M may be dependent on $Te(i)$, it considers expressing M by $Te(i)$. It will be set to $M \leq \text{ceiling}(Te(i)/Tv_{min}) \leq Te(i)/Tv_{min}+1$ if the minimum of VU regeneration time of RU(i) Naka is set to Tv_{min} . $\text{ceiling}(x)$ is a function which searches for the minimum integer more than x .

[0112]At this time, it is $< \text{type} \text{ It is } < \text{type even if it substitutes } Te(i)/Tv_{min}+1 \text{ for } M \text{ of } 4> \text{ If } Te(i) \text{ is set up so that } 1> \text{ may be materialized, if VU regeneration time is more than } Tv_{min}, \text{ even if it constitutes RU from VU of what kind of regeneration time, real-time postrecording will be attained.}$

[0113] $< \text{type It is } < \text{type to } 1> \text{ 3}> \text{ and } < \text{type When } 4> \text{ is substituted and it solves by } Te(i), \text{ it is condition } Te(i) \geq 3Ta/(Rs-Rv-Ra-Rp-(Tk/Tv) \times Rs-2Ly/Tv) \text{ of } Te(i) \text{ which can guarantee real-time postrecording... } < \text{type } 5> \text{ is obtained.}$

[0114]That is, RU regeneration time lower limit $Temin$ in which a postrecording guarantee is possible is $Temin=3Ta/(Rs-Rv-Ra-2Rp-(Tk/Tv_{min}) \times Rs-2Ly/Tv) \dots < \text{type It becomes } 6> \text{.}$

[0115]At this time, the upper limit $Temax$ of RU regeneration time is set up as follows.
 $Temax=Temin+Tv_{max} \dots < \text{type } 7> \text{ Here, } Tv_{max} \text{ is the maximum regeneration time of VU. Setting up upper limit is based on the following reason. At the time of postrecording, periods from (2) to (8) of drawing 8 become long as } Te \text{ becomes large. In the meantime, since read-out from a disk of data for playback cannot be performed, in order to continue playback, according to an increase in } Te, \text{ it is necessary to increase size of the track buffer 502. Upper limit is set up in order to enable an estimate of size of the track buffer 502 which is needed at this time.}$

[0116]When a margin for the maximum regeneration time of VU is between a lower limit and upper limit, it is possible to constitute RU from combination of VU of arbitrary regeneration time. Although the maximum regeneration time is set up here according to the bit rate of an AV stream, based on the greatest possible bit rate, it is not concerned with the bit rate of an AV stream, but is good also as fixed.

[0117]Although a continuous recording unit is managed by regeneration time in this example, it cannot be overemphasized that it may manage in record section size which multiplied regeneration time by the bit rate of data.

[0118]Although an ECC block including a PRU boundary read at the time of data read for reproduction is transmitted to the buffer 504 for postrecording and used from the track buffer 502 in this example at the time of PRU record, an ECC block which does not transmit but includes a PRU boundary just before PRU record may be read.

[0119]In this case, PRU reading time and pickup transit time will take too much in the case of PRU record. every — a case where an ECC block including a PRU boundary is read just before record about PRU — $< \text{type } Tw(i)=2Ta+(M-1) \times Tk+M \times Tr+TexRp/Rs+2x(2M-1) \times Ly/Rs \dots Tw(i) \text{ of } 1> \text{ — } < \text{type What is necessary is just to calculate by } 8> \text{. } Tr \text{ is the maximum latency speed.}$

[0120]In this example, it assumes moving a pickup to RU of a division jump and the past asynchronously. Since this reason is severe as conditions for performing real-time postrecording compared with a case where it carries out by a direction performed asynchronously synchronizing (a period when read-out of data for reproduction breaks off is long), It is because it is asynchronous, a synchronization is also possible if real-time postrecording is possible, and it is possible to raise flexibility of mounting.

[0121]Therefore, $Temin$ may be set up on the assumption that movement of a pickup to RU of a division jump and the past is performed synchronously. In this case, $< \text{type What is necessary is to remove the 2nd paragraph of } 3> \text{ and just to think.}$

[0122]Next, an arrangement deciding method on a disk of a stream corresponding to un-[postrecording] is explained. It is $< \text{type like a stream corresponding to postrecording. Seamless reproduction will be guaranteed if } 1> \text{ is filled. However, it is set to } Twa=0 \text{ in order not to perform record of postrecording data.}$

[0123] $Tr(i) \text{ is } < \text{type. With } 3>, \text{ it is common, comes out, and, for a certain reason, is } < \text{type. It is } < \text{type to } 1> \text{. When } 3> \text{ and } Twa=0 \text{ is substituted and it solves by } Te(i), \text{ it is } Te(i) \geq Ta/(Rs-Rv-Ra) \dots < \text{type } 9> \text{ is obtained.}$

[0124]That is, RU regeneration time lower limit $Temin$ in which a seamless reproduction guarantee is possible is $Temin \geq Ta/(Rs-Rv-Ra) \dots < \text{type It becomes } 10> \text{.}$

[0125]That is, the ranges of RU regeneration time will differ between a stream and a stream corresponding to un-corresponding to postrecording. Because, when a stream corresponding to un-is made into the same range as a correspondence stream, it is $< \text{type on a disk. It is } < \text{type although } 10> \text{ is filled. It is because a choice of recording a stream corresponding to un-[postrecording] will be lost when there is an empty continuation field which does not fill } 6> \text{.}$

[0126]<Buffer size>, next size of the track buffer 502 required at the time of postrecording are explained. Here, it does not take into consideration about disturbance, such as a shock. Required size increases, so that dispersion in an interval of timing which shifts from regenerative data read-out to PRU record is large. For example, since consumption of data in the track buffer 502 will occur continuously if it seems that a case where an interval of timing from which it shifts to PRU record is narrow continues, more capacity is required for the track buffer 502.

[0127]Here, it thinks from a postrecording data recording preparation completion of RU#N which is arbitrary RUs by a case which is RU#N+1 which is the following RU where it is said that postrecording data recording processing ((2) - (8) of drawing 8) of RU#N will be completed by the time it carries out a postrecording data recording preparation completion. This width is a start possible value about postrecording data recording, after read-out of RU which is performing the present read-out is completed.

[0128]On this condition, it becomes [in which timing of a before / from a postrecording data recording preparation completion of a certain RU / a postrecording data recording preparation completion of the following RU / a division jump and postrecording data recording occur, and] unfixed. Therefore, so that reproduction may not break off, even if one postrecording data recording and one division jump occur continuously immediately after a postrecording data recording preparation completion, Even if it

performs continuously one postrecording data recording and one division jump at the time of a postrecording data recording preparation completion, there must be data in which continuous reproduction is possible in the track buffer 502 (conditions 1). [0129]It is also considered that postrecording data recording and a division jump occur just before a postrecording data recording preparation completion. In order to fulfill the above-mentioned conditions 1 also in such a case, The track buffer 502 must be able to hold data in which continuous reproduction is possible, even if two postrecording data recording and two division jumps occur continuously just before one postrecording data recording in front of a postrecording data recording preparation completion, and one division jump (conditions 2).

[0130]That is, capacity corresponding to regeneration time of two postrecording data recording and division jump 2 batch in the track buffer 502 is needed. In addition, even if it is the regeneration time T_{tmax} required in order to carry out continuous reproduction until it reads following VU, and the combination of arbitrary VU, including the margin T_{vmax} for securing required regeneration time to a track buffer, the size L_t of the required track buffer 502 is $L_t \geq (T_{\text{tmax}} + 2(T_{\text{wmax}} + T_a) + T_{\text{vmax}}) \times (R_v + R_a + R_p) \dots$ < type It becomes 11>.

[0131]Here $T_{\text{tmax}} = T_{\text{vmax}} \times (R_v + R_a + R_p) / R_s$, About $T_{\text{wmax}} = 2 T_a + T_{\text{max}} \times T_k / (T_{\text{vmin}} + (2 T_{\text{max}} \times L_y) / (T_{\text{vmin}} \times R_s) + T_{\text{max}} \times R_a / R_s)$, it is < type. It substitutes for 11>, $L_t \geq (T_{\text{vmax}} \times (R_v + R_a + R_p) + 2(3 T_a + (T_{\text{max}} \times T_k) / T_{\text{vmin}}) \times R_s + 2 T_{\text{max}} \times L_y / T_{\text{vmin}} + T_{\text{max}} \times R_p) \times (R_v + R_a + R_p) / R_s \dots$ < type 12> is obtained.

[0132]Next, size of the buffer 504 for postrecording which is needed at the time of postrecording is explained. Influence is received in dispersion in an interval of timing which shifts from regenerative data read-out to PRU record as well as track buffer size also about the buffer 504 for postrecording.

[0133]Here, like track buffer size trial calculation from a postrecording data recording preparation completion of RU#N which is arbitrary RUs. It thinks by a case which is RU#N+1 which is the following RU where it is said that postrecording data recording processing (2) - (8) of drawing 8) of RU#N will be completed by the time it carries out a postrecording data recording preparation completion.

[0134]On this condition, PRU record for the two maximum RUs may continue. A PRU field for one RU for storing audio information encoded in the midst of PRU record is needed. Therefore, there should just be a field which stores a PRU block for three RUs.

[0135]Therefore, the size L_p of the buffer 504 for postrecording is $L_p \geq 3 \times (T_{\text{max}} \times R_p + (2 \times (T_{\text{max}} / T_{\text{vmin}}) - 1) \times L_y) \dots$ < type It becomes 13>. The 2nd paragraph of the right-hand side is for holding an ECC block including a PRU boundary.

[0136]Next, size of the track buffer 502 required at the time of reproduction of a stream corresponding to un-[postrecording] is explained. Here, it does not take into consideration about disturbance, such as a shock.

[0137]That a track buffer is most needed is a case where a division jump occurs during read-out of VU of the regeneration time T_{vmax} . Since new VU is not read to a track buffer in the meantime, it is necessary to continue reproduction by data in a track buffer.

[0138]That is, regeneration time for $T_{\text{rmax}} + T_a$ must exist in the track buffer 502. Here, T_{rmax} expresses reading time of VU of the regeneration time T_{vmax} . At this time, the size L of the required track buffer 502 is $L \geq (T_{\text{rmax}} + T_a + T_{\text{vmax}}) \times (R_v + R_a) + T_{\text{vmax}} \times (R_v + R_a) \dots$ It becomes <the formula 14>.

[0139]The 1st paragraph of the right-hand side is data volume required in order to reproduce until it reads new VU. Even if regeneration time is a case where VU of what kind of regeneration time is intermingled in the track buffer 502, by T_{vmax} in many one, it is for certainly securing required regeneration time into the track buffer 502.

[0140]The 2nd paragraph of the right-hand side is a field for reading and holding inner VU. In addition, although a margin for intermittent reproduction at the time of buffer full is required, it omits here.

[0141]Here, $T_{\text{rmax}} = T_{\text{vmax}} \times (R_v + R_a) / R_s$ is substituted for <the formula 14> and it is $L \geq (T_{\text{vmax}} \times (R_v + R_a + 2 R_s) + T_a \times R_s) \times (R_v + R_a) / R_s \dots$ <The formula 15> is obtained.

[0142]Processing when recording is directed is explained along with a flow chart of drawing 9 from a <processing at time of record> user. AV streams recorded at this time are bit rate $R_v = 5\text{Mbps}$ of video, 48 kHz of sampling frequencies of an audio, and bit rate $R_a = 256\text{k bps}$, and presuppose that it is a stream corresponding to postrecording of VU regeneration time immobilization. Management information of a file system shall already be read on RAM.

[0143]First, it opts for composition of a stream, or composition of a continuation field (Step 701). It is < type when 1 VU is constituted from 15 1GOP. 6>, < type $R_v = 5\text{Mbps}$, $R_a = R_p = 256\text{k bps}$, and $T_{\text{vmax}} = T_{\text{vmin}} =$ about 0.5 second are substituted for 7> for $T_k = 0.1$ second for $R_s = 20\text{Mbps}$, and $T_a = 1$ second, and less than the more than range 10.49 second 10.99 second of $T_e(i)$ is obtained.

[0144]In MPEG-1 audio layer-II which it becomes the time of $T_e(i) = 10.5$ seconds to fulfill this condition in $T_{\text{vmax}} =$ about 0.5 second, and will constitute RU 21 whole VU. Since regeneration time per AAU will be 0.024 second when a sampling frequency is 48 kHz, 20 pieces or 21 AAU(s) go into 1 VU.

[0145]When a refreshable maximum bit rate of an audio is 256k bps, the maximum size of each AAU is set to 768 bytes. Therefore, area size of each PRU is secured by AAU number $\times 768\text{byte}$ of a corresponding main audio.

[0146]A reason which makes area size of PRU a fixed value is shown below. When recording a stream corresponding to postrecording, the area size of PRU stores same number as an AAU number contained in corresponding VU of AAU(s).

[0147]However, it is difficult to get to know strictly [before encoding an AAU number contained in corresponding VU]. Because, it is because a cycle of AAU and GOP does not become an integral multiple and an audio in VU and start timing of video are not necessarily in agreement. Therefore, area size of PRU is taken as the maximum AAU number contained in regeneration time $T_e(i)$.

[0148]Although the whole bit rate increases by this, when audio bit rate R_a is 256k bps, it is a maximum of 768 bytes, and may ignore at several seconds. Although data which is not contributed to reproduction will be contained in an AV stream, since it is excludable from a reproduction object using QuickTime management information, it is satisfactory.

[0149]Next, continuously recordable free space is looked for for 21 VU. Specifically, $21 \times T_{\text{vmax}} \times (R_v + R_a + R_p)$, i.e., continuous free space of 57.9 or more Mbit, is looked for with reference to Space Bitmap on RAM102. If it does not exist, a user is told about the ability to stop recording and not record it (Step 702).

[0150]Next, the audio encoder 117 and the video encoder 118 are started, respectively (Step 703). It confirms whether the above data is stored in a buffer for record by 1 ECC block (32 KB) (Step 704), and while being accumulated, Step 708 is repeated from Step 705.

[0151]If accumulated, empty situations of an ECC block on a disk recorded on the next will be investigated with reference to

Space Bitmap on RAM (Step 705). If there is no opening, continuous free space which can record 21 VU will be looked for (Step 707), a pickup will be moved to a head of the free space (Step 708), and data for 1 ECC block in the buffer 111 for record will be recorded on a disk (Step 706).

[0152] If data is not stored in the buffer 111 for record, it confirms whether an end of record is directed (Step 709), and Step 704 will be performed if it is not the end of record.

[0153] The following steps are performed when an end of record is directed. First, about data which is less than 32 KB in a buffer for record, dummy data is added to an end and it is made 32 KB (Step 710). Next, the data is recorded on a disk (Step 711 – Step 714). Finally, QuickTime management information and file system management information on RAM102 are recorded on the optical disc 106 (Step 715 – Step 716).

[0154] Operation of the audio encoder 117 and the video encoder 118 which are parallel with the above processing, or the multiplexer 113 is explained. Each encoder sends an encoding result to the multiplexer 113, and a multiplexer stores them in the buffer 114 for multiplexing.

[0155] If data for 1 VU, i.e., 1GOP, and AAU reproduced synchronizing with it are accumulated in the buffer 114 for multiplexing, the multiplexer 113 will send 1-VU data to the buffer 111 for record. In that case, the multiplexer 113 multiplexes PRU which can store AAU of a maximum bit rate according to the number of AAU in the VU.

[0156] Reporting that data for 1 VU has been encoded to the host CPU 101, the host CPU 101 updates QuickTime management information on RAM102 based on a number and size of GOP or AAU which constitute VU.

[0157] Here, to a recorded stream, based on performance (Rs, Ta, Tk) of a reference device, regeneration time included in RU was determined so that it could postrecord by any apparatus, but this may be determined based on performance which picture recording apparatus has. In that case, performance is stored in set performance atom of X descriptor atom in QuickTime management information so that it can be judged whether apparatus which postrecords can postrecord to the AV stream.

[0158] Processing when postrecording is directed is explained along with a flow chart of drawing 10 from a <processing at time of postrecording> user. Here, QuickTime management information about an AV stream which is already the target of postrecording shall be read into RAM102.

[0159] First, it is reported that it investigates whether the QuickTime movie comprises a stream corresponding to postrecording of only one file, otherwise, cannot postrecord to a user (Step 801). What carried out non-destroying edit of the streams by which this was independently recorded on a disk is because there is no guarantee which fulfills conditions for postrecording mentioned above.

[0160] Data for playback is read from a head of VU on a disk including a postrecording starting position (Step 802). At this time, Step 802 is repeated until it reads data for sufficient regeneration time (Step 803). Here, data for sufficient regeneration time means only data volume in which reproduction does not break off, even when an interruption period of data read for reproduction is the maximum.

[0161] When PRU is read, an ECC block containing PRU is sent to the buffer 111 for postrecording. In order to manage PRU in the buffer 111 for postrecording at this time, it holds to RAM102 by using a group of reproduction time of onset (relative time from a head of an AV stream) of each PRU in the buffer 111 for postrecording, and an address in inside of the buffer 111 for postrecording as a table.

[0162] Next, the video decoder 116, the audio decoder 115, and the audio encoder 117 are started (Step 804). The audio encoder 117 encodes a sampled voice waveform to AAU, and sends it to the multiplexer 113 with a cycle of AAU. In that case, relative time from a head of an AV stream is added about each AAU.

[0163] The multiplexer 113 stores AAU in PRU in the buffer 111 for postrecording based on time added to AAU. If it finishes storing AAU in PRU of the last in RU to the last, an end of encoding of RU will be notified to the host CPU 101.

[0164] Next, it is confirmed whether an end of postrecording is directed from a user (Step 805). Data for reproduction is read like Step 802 until encoding of PRU is completed, if not directed (Step 809).

[0165] If an end of RU encoding is notified from a multiplexer part (Step 806). It asks for an address on the optical disc 106 which should record the PRU using QuickTime management information, i.e., an address with which the PRU was recorded from the first, from playback time of onset of PRU contained in the RU currently held on a table on RAM102. The pickup 107 is moved to the address (Step 807), and an ECC block containing the PRU is recorded on the optical disc 107 (Step 808).

[0166] If an end of postrecording is directed, it will wait for the completion of encoding of PRU under present encoding (Step 810), a pickup will be moved in quest of a recording address of the PRU (Step 811), and PRU will be recorded (Step 812). Finally QuickTime management information is recorded on a disk (Step 813).

[0167] In this example, when interrupting read-out of data for reproduction and starting record of PRU, an occupation of the buffer 110 for reproduction is not checked, but in order to improve tolerance to a shock at the time of postrecording, it cannot be overemphasized that it is better to check. However, capacity of a part which is in PRU record timing in this case, more buffers 110 for reproduction, and the buffer 111 for postrecording is needed.

[0168] Processing when reproduction is directed is explained along with a flow chart of drawing 11 from a <processing at time of reproduction> user. QuickTime management information about an AV stream which is already the reproductive target here shall be read into RAM102.

[0169] Data for playback is read from a head of VU on the optical disc 107 by which reproduction instruction was carried out (Step 901). At this time, Step 901 is repeated until it reads data for sufficient regeneration time (Step 902).

[0170] Here, data for sufficient regeneration time means only data volume in which reproduction does not break off, even when an interruption period of data read for reproduction is the maximum. Specifically, it is considered as data volume for 1 second supposing a case where division accompanying read-out of AV information is jumped (a maximum of 1 second).

[0171] Next, the video decoder 116 and the audio decoder 115 are started (Step 903). It is confirmed whether an end of reproduction is directed from a user (Step 904). If not directed, AV information for reproduction is read (Step 905). It will end, if an end of reproduction is directed.

[0172] The <2nd example>, next the 2nd example in this invention are described. A difference from the 1st example mentioned above is the point of rewriting the whole RU by this example to the 1st example rewriting only PRU. Since this example is similar with the 1st example, it is extracted in a point of difference and described below. A definition in the 1st example is used for a sign which is not newly defined.

[0173] Since composition of an AV stream in <gestalt of AV stream> this example is completely the same as that of the 1st above-mentioned example, explanation is omitted.

[0174]A deciding method of RU regeneration time in a stream corresponding to un-[<disk arrangement deciding method> postrecording] is as common as the 1st above-mentioned example.

[0175]Next, a deciding method of RU regeneration time in a stream corresponding to postrecording is explained. Like the 1st example, supposing a reference device model and a reference postrecording algorithm, when it postrecords using them, in this deciding method, RU regeneration time is decided that seamless reproduction does not fail for compatibility reservation between apparatus.

[0176]About a reference device model, the same composition as a thing of the 1st example mentioned above with drawing 7 is taken. However, in the 2nd example, the whole RU is transmitted to being only an ECC block in which being transmitted to the buffer 504 for postrecording from the track buffer 502 in the 1st example contains PRU. Not only an ECC block that contains PRU at the time of record on a disk of postrecording data but the whole RU is recorded.

[0177]Next, a reference postrecording algorithm is explained using drawing 12. Numbers from (1) to (5) in drawing 12 are equivalent to numbers from (1) to (5) under following explanation. The outline of an algorithm is as follows.

[0178](1) Read data for reproduction. (2) Perform access to RU#N at the same time encoding of audio information corresponding to RU#N which is Nth RU is completed. (3) Record RU#N for data of the buffer 504 for postrecording. (4) Return to the original reading position. (5) Resume read-out of data for reproduction. The above operation is repeated.

[0179]When it postrecords in said reference device model using said reference postrecording algorithm, it is < type like the 1st example. By filling 1>. It can guarantee that there are not overflow of the buffer 504 for postrecording and underflow of the track buffer 502.

[0180]In the 2nd example, it is < type. $Tr(i)$ in 1> is < type. It is as common as 3>. $Tw(i)$ in <the formula 1> is $Tw(i)=2 Ta+Te(i) \times (Rv+Ra+Rp)/Rs$... < type 16> Here, the 1st paragraph of the right-hand side shows both-way access time to RU#i. The 3rd paragraph of the right-hand side expresses time for recording RU#i on a disk. Since it comprises an ECC block of integer pieces, RU is < type. A paragraph like the 4th paragraph of the right-hand side of 4> is unnecessary.

[0181]< type It is < type to 1>. 3> and < type When 16> is substituted and it solves by $Te(i)$, it is condition $Te(i) \geq 3Ta/(Rs-2(Rv+Ra+Rp))$ of $Te(i)$ which can guarantee real-time postrecording... < type 17> is obtained.

[0182]That is, RU regeneration time lower limit $Temin$ in which a postrecording guarantee is possible is $Temin=3Ta/(Rs-2(Rv+Ra+Rp))$... < type It becomes 18>.

[0183]At this time, the upper limit $Temax$ of RU regeneration time is set up as follows.

$Temax=Temin+Tvmx$... < type A reason for setting upper limit as 19>RU regeneration time is the same as a reason explained in the 1st example. As the 1st example explained, it is not concerned with the bit rate of an AV stream based on a possible maximum bit rate, but is good as fixed in the maximum regeneration time.

[0184]Although a continuous recording unit is managed by regeneration time in this example, it cannot be overemphasized that it may manage in record section size which multiplied regeneration time by the bit rate of data.

[0185]Although it assumes moving a pickup to RU of a division jump and the past asynchronously in this example, it may assume that it carries out synchronously like the 1st example, and $Temin$ may be set up.

[0186]<Track buffer size>, next size of the track buffer 502 required at the time of postrecording are explained. Here, it does not take into consideration about disturbance, such as a shock. Like the 1st example, required size increases, so that dispersion in an interval of timing which shifts from regenerative data read-out to PRU record is large.

[0187]Here, a case which is RU#N+1 which is the following RU where it is said that postrecording data recording processing ((2) - (4) of drawing 12) of RU#N will be completed by the time it carries out a postrecording data recording preparation completion is considered like the 1st example from a postrecording data recording preparation completion of RU#N which is arbitrary RUs. Unlike the 1st example, postrecording data recording here refers to record of the whole RU.

[0188]In this case, the track buffer 502 serves as capacity corresponding to regeneration time of two postrecording data recording and division jump 2 batch for the reason as explanation given in the 1st example capacity of a required track buffer is the same. That is, the size Lt of the required track buffer 502 is < type. It is expressed with 11>.

[0189]Here $Ttmax=Tvmx \times (Rv+Ra+Rp)/Rs$, $Twmax=2 Ta+Temax \times (Rv+Ra+Rp)/Rs$ is substituted for <the formula 11>, and it is $L=(2 Temax+Tvmx) \times (Rv+Ra+Rp) \times (Tvmx+6Ta) \times Rs / (Rv+Ra+Rp) \times Rs$... < type 20> is obtained.

[0190]Next, size of the buffer 504 for postrecording which is needed at the time of postrecording is explained. For a reason explained in the 1st example, size of the buffer 504 for postrecording is needed by 3RUs. Therefore, the size Lp of the buffer 504 for postrecording is $Lp \geq 3 \times Temax \times (Rv+Ra+Rp)$... < type It becomes 21>.

[0191]At the time of record, since it is the same as that of the 1st example, processing at the time of reproduction is omitted. Processing at the time of postrecording differs from the 1st example the following point. First, in Step 803, in the 1st example, only when PRU is read, a data transfer from the buffer 110 for reproduction to the buffer 111 for postrecording is performed, but by this example, it transmits about all the read data.

[0192]Next, although only an ECC block containing PRU is recorded on a disk in the 1st example in Step 808, it records about all VU for postrecording in this example.

[0193]RU reproduction lower limit and < type which are obtained by <the formula 17> in a system to carry out although it is proper use with this example and the 1st example RU reproduction lower limit obtained by 9> is computed, and RU reproduction lower limit should just adopt a direction which becomes small. Because, the one where RU is shorter is hard to attach restriction of arrangement etc.

[0194]The <3rd example> In the 1st and 2nd examples mentioned above, although premised on distributing and recording an AV stream on a disk per RU, in apparatus with long pickup transit time, a case where postrecording is impossible can be considered by these methods.

[0195]The 3rd example in this invention is making it continuously arranged on a disk only within an AV stream of postrecording correspondence in such a case, reduces maximum time of pickup transit time, and postrecords possible also by apparatus with long pickup transit time.

[0196]About a gestalt of an AV stream, it is as common as the 1st above-mentioned example. However, about an AV stream corresponding to postrecording, one AV stream comprises one RU.

[0197]Record of an AV stream corresponding to <processing at time of record> postrecording is performed by the same method as the 1st example. However, in the 1st example, when record of continuously recordable free space is completed, when the 1st record of continuously recordable free space is completed, it differs in that record is ended to moving to the next free space and continuing record, at this example.

[0198]<Processing at time of postrecording> postrecording processing is performed in the same procedure as the 1st example. However, since there is no unit of RU, timing which starts postrecording data recording is determined by one method of the following.

[0199]First, the 1st method is explained. The 1st method is < type on the assumption that only an ECC block containing PRU is recorded like the 1st example. Postrecording data recording is started at intervals of Temin obtained by 6>.

[0200]However, in the 1st example, although Ta was the greatest time that read-out from a disk and record stop, by this example, it becomes what doubled seek time and latency speed of about hundreds of tracks from tens tracks.

[0201]Next, the 2nd method is explained. Like the 2nd example, the 2nd method is < type on the assumption that main audios other than a postrecording audio and a video data are also rewritten. Postrecording data recording is started at intervals of Temin obtained by 13>.

[0202]However, in the 2nd example, although Ta was the greatest time that read-out from a disk and record stop, by this example, it becomes what doubled seek time and latency speed of about hundreds of tracks from tens tracks.

[0203]The <4th example> In an example of the 1st thru/or 3 mentioned above, although premised on performing audio information record on a disk in parallel to an audio input at the time of postrecording, when it is apparatus with long pickup transit time, it is difficult to perform such control. The 4th example of this invention assumes such a case.

[0204]About a gestalt of an AV stream, it is as common as the 1st above-mentioned example. Regeneration time per RU is related with a stream corresponding to postrecording, and is < type like a stream corresponding to un-[postrecording]. It determines based on 10>. Since it is the same as that of the 1st example, processing at the time of record and reproduction is omitted.

[0205]Processing when postrecording is directed is explained along with a flow chart of drawing 13 from a <processing at time of postrecording> user. Here, QuickTime management information about an AV stream which is already the target of postrecording shall be read into RAM102.

[0206]First, data for playback is read from a head of VU on a disk including a postrecording starting position (Step 1301). At this time, Step 1301 is repeated until it reads data for sufficient regeneration time (Step 1302). Here, data for sufficient regeneration time means only data volume in which reproduction does not break off, even when an interruption period of data read for reproduction is the maximum.

[0207]When PRU is read, an ECC block (PRU block) containing PRU is sent to the buffer 111 for postrecording. In order to manage PRU in the buffer 111 for postrecording at this time, it holds to RAM102 by using a group of reproduction time of onset (relative time from a head of an AV stream) of each PRU in the buffer 111 for postrecording, and an address in inside of the buffer 111 for postrecording as a table.

[0208]Next, the video decoder 116, the audio decoder 115, and the audio encoder 117 are started (Step 1303). The audio encoder 117 encodes a sampled voice waveform to AAU, and sends it to the multiplexer 113 with a cycle of AAU.

[0209]In that case, relative time from a head of an AV stream is added about each AAU. The multiplexer 113 stores AAU in a PRU block in the buffer 111 for postrecording based on time added to AAU.

[0210]Next, it is confirmed whether an end of postrecording is directed from a user (Step 1304). Data for reproduction is read like Step 1301 until encoding of PRU is completed, if not directed (Step 1305). The following processings are performed as long as an unrecorded PRU block is in the buffer 111 for postrecording (Step 1306), if an end of postrecording is directed.

[0211]First, it asks for an address on the optical disc 106 which should record the PRU block using QuickTime management information, i.e., an address with which the PRU block was recorded from the first, from playback time of onset of the PRU currently held on a table on RAM102.

[0212]Next, the pickup 107 is moved to the address (Step 1307), and an ECC block containing the PRU is recorded on the optical disc 107 (Step 1308). If an unrecorded PRU block is lost, finally QuickTime management information will be recorded on a disk (Step 1309).

[0213]In the above postrecording processing, after postrecording all the PRU(s), it is recording, but PRU recordable during postrecording is recording and it is possible to reduce waiting time after postrecording. Such processing is explained along with a flow chart of drawing 14. QuickTime management information about an AV stream which is already the target of postrecording here shall be read into RAM102.

[0214]First, data for playback is read from a head of VU on a disk including a postrecording starting position (Step 1401). At this time, Step 1401 is repeated until it reads data for sufficient regeneration time (Step 1402). Here, data for sufficient regeneration time means only data volume in which reproduction does not break off, even when an interruption period of data read for reproduction is the maximum.

[0215]When PRU is read, an ECC block (PRU block) containing PRU is sent to the buffer 111 for postrecording. In order to manage PRU in the buffer 111 for postrecording at this time, it holds to RAM102 by using a group of reproduction time of onset (relative time from a head of an AV stream) of each PRU in the buffer 111 for postrecording, and an address in inside of the buffer 111 for postrecording as a table.

[0216]Next, the video decoder 116, the audio decoder 115, and the audio encoder 117 are started (Step 1403). The audio encoder 117 encodes a sampled voice waveform to AAU, and sends it to the multiplexer 113 with a cycle of AAU. In that case, relative time from a head of an AV stream is added about each AAU.

[0217]The multiplexer 113 stores AAU in a PRU block in the buffer 111 for postrecording based on time added to AAU. If it finishes storing AAU in PRU of the last in RU to the last, an end of encoding of RU will be notified to the host CPU 101.

[0218]Next, it is confirmed whether an end of postrecording is directed from a user (Step 1404). Data for reproduction is read like Step 1301 until encoding of PRU is completed, if not directed (Step 1409).

[0219]If an end of PRU encoding is notified from a multiplexer part (Step 1405) It is judged whether the PRU is recordable (Step 1406). The trial calculation of time which the PRU record takes from a position and physical relationship of PRU of the present pickup is specifically made, and as compared with the present buffer residue for reproduction, if a buffer for reproduction does not become close to empty or empty by PRU record, it is judged that record is possible.

[0220]When it is judged that record is possible, it asks for an address on the optical disc 106 which should record the PRU using QuickTime management information, i.e., an address with which the PRU was recorded from the first, from playback time of onset of PRU contained in the RU currently held on a table on RAM102. The pickup 107 is moved to the address (Step 1407), and an ECC block containing the PRU is recorded on the optical disc 107 (Step 1408).

[0221]The following processings are performed as long as an unrecorded PRU block is in the buffer 111 for postrecording (Step

1410), if an end of postrecording is directed. First, it asks for an address on the optical disc 106 which should record the PRU block using QuickTime management information, i.e., an address with which the PRU block was recorded from the first, from playback time of onset of the PRU currently held on a table on RAM102.

[0222]Next, the pickup 107 is moved to the address (Step 1411), and an ECC block containing the PRU is recorded on the optical disc 107 (Step 1412). If an unrecorded PRU block is lost, finally QuickTime management information will be recorded on a disk (Step 1413).

[0223]The <5th example> Although an example of the 1st thru/or 3 mentioned above is recording directly audio information inputted at the time of postrecording on corresponding PRU, it may be difficult to perform such control in apparatus with long pickup transit time. Supposing such a case, the 5th example in this invention records on a temporary field on a disk at the time of postrecording, and moves to original PRU after postrecording.

[0224]About a gestalt of an AV stream, it is as common as the 1st above-mentioned example. Regeneration time per RU is related with a stream corresponding to postrecording, and is < type like a stream corresponding to un-[postrecording]. It determines based on 10>. Since it is the same as that of the 1st example, processing at the time of record and reproduction is omitted.

[0225]Processing when postrecording is directed is explained along with a flow chart of drawing 15 from a <processing at time of postrecording> user. Here, QuickTime management information about an AV stream which is already the target of postrecording shall be read into RAM102.

[0226]First, data for playback is read from a head of VU on a disk including a postrecording starting position (Step 1501). At this time, Step 1501 is repeated until it reads data for sufficient regeneration time (Step 1502). Here, data for sufficient regeneration time means only data volume in which reproduction does not break off, even when an interruption period of data read for reproduction is the maximum.

[0227]When PRU is read, an ECC block (PRU block) containing PRU is sent to the buffer 111 for postrecording. In order to manage PRU in the buffer 111 for postrecording at this time, it holds to RAM102 by using a group of reproduction time of onset (relative time from a head of an AV stream) of each PRU in the buffer 111 for postrecording, and an address in inside of the buffer 111 for postrecording as a table.

[0228]Next, the video decoder 116, the audio decoder 115, and the audio encoder 117 are started (Step 1503). The audio encoder 117 encodes a sampled voice waveform to AAU, and sends it to the multiplexer 113 with a cycle of AAU. In that case, relative time from a head of an AV stream is added about each AAU.

[0229]The multiplexer 113 stores AAU in a PRU block in the buffer 111 for postrecording based on time added to AAU. The multiplexer 113 stores AAU in PRU in the buffer 111 for postrecording based on time added to AAU.

[0230]Next, it is confirmed whether an end of postrecording is directed from a user (Step 1504). If not directed, the pickup 107 is moved to following VU (Step 1505), and data for reproduction is read like Step 802 (Step 1506).

[0231]When a certain PRU block is after [from a postrecording start] the Nth piece in the buffer 111 for postrecording now (Step 1507), The pickup 107 is moved to a head of VU read immediately before (Step 1508), and the oldest PRU block in the buffer 111 for postrecording is recorded temporarily (Step 1509).

[0232]It records on RAM102 which PRU was temporarily recorded where on that occasion. Above-mentioned N determines postrecording data to the data for reproduction based on the greatest time that becomes recordable, after reading data for reproduction.

[0233]The next processing is performed as long as PRU recorded temporarily remains (Step 1510), if an end of postrecording is directed. First, the pickup 107 is moved to a position of PRU recorded temporarily (Step 1511). A PRU block is read (Step 1512), the pickup 107 is moved to a position of PRU which should record the PRU essentially (Step 1513), and a PRU block is recorded (Step 1514).

[0234]Although written for every PRU here, it cannot be overemphasized that read-out and record may be continuously performed about two or more PRU(s).

[0235]The next processing is performed, as long as a PRU block remains in the buffer 111 for postrecording if movement at an original place of PRU recorded temporarily finishes next (Step 1515). It asks for an address with which the PRU block was recorded from the first using QuickTime management information from reproduction time of onset of PRU added to a PRU block. The pickup 107 is moved to the address (Step 1516), and the PRU block is recorded (Step 1517). If an unrecorded PRU block is lost, finally QuickTime management information will be recorded on a disk (Step 1518).

[0236]Although another field in the same stream is used in consideration of a case where there is no free space in a disk, in this example as a field which records a PRU block temporarily, if it is near the present reading position, it is clear that it may be outside the same stream.

[0237]In this example, although semiconductor memory and an optical disc like a buffer for postrecording are used as a medium which records a PRU block temporarily, it cannot be overemphasized that you may be the other recording media, such as a hard disk.

[0238]The <6th example>, next the 6th example in this invention are described with drawing 16.

[0239]<a gestalt of an AV stream> -- composition of an AV stream in this example is first explained using drawing 16. There are a stream and a stream corresponding to un-[postrecording] corresponding to postrecording in an AV stream. A stream corresponding to un-[postrecording] comprises a unit which multiplexed video and an audio for about 1 second. Here, the unit is called Video Unit (VU).

[0240]Here, about composition of VU, since it is common, explanation is abbreviated to VU in a stream corresponding to un-[postrecording] in the 1st example mentioned above.

[0241]On the other hand, a stream corresponding to postrecording multiplexes Post Recording Unit (PRU) which is a field for storing postrecording data for every VU of integer pieces to a stream corresponding to un-[postrecording], as shown in drawing 16. Audio information for postrecording reproduced at the same time as VU of a before [the next PRU] is stored in PRU.

[0242]It is located before VU to which PRU corresponds, that is, a thing with a small relative block number of a under [a file] is because there are few buffers for delay at the time of carrying out synchronous reproduction of AV information in VU and the audio information for postrecording in PRU and they end.

[0243]Here, it is referred to as Editable Unit (EU) in accordance with PRU and VU of integer pieces by which synchronous reproduction is carried out. An interval which multiplexes PRU is connected with the implementability of postrecording, and a

deciding method of a multiplexing interval is later mentioned for it.

[0244]Inside of EU (the EU head is also included.) However, division which removes a head of an AV stream shall consider it as a maximum of 1 time, and PRU shall be continuously recorded on a disk. By this, a minimum of the EU regeneration time can be lowered so that it may mention later.

[0245>About a deciding method of area size of PRU, since it is common, explanation is abbreviated to the 1st example.

[0246]A <disk arrangement deciding method>, next a deciding method of an interval which multiplexes PRU in a stream corresponding to postrecording are explained. A device (reference device model) which serves as a standard like the 1st example in this deciding method. When it postrecords by next using them supposing a postrecording algorithm (reference postrecording algorithm) used as a standard, a multiplexing interval is decided that seamless reproduction does not fail.

[0247]Here, about a reference device model, since it is common, explanation is abbreviated to the 1st example.

[0248]Next, a reference postrecording algorithm is explained using drawing 17. Numbers from (1) to (11) in drawing 17 are equivalent to numbers from (1) to (11) under following explanation. The outline of an algorithm is as follows.

[0249](1) Read data for reproduction. (2) Perform access to PRU (N) at the same time encoding of audio information corresponding to PRU (N) which is Nth PRU is completed. (3) Record PRU (N) on a disk. (4) Return to the original reading position.

[0250](5) Read data for reproduction. (6) Perform access to PRU (N+1) at the same time encoding of audio information corresponding to PRU (N+1) which is the N+1st PRU(s) is completed. (7) Record PRU (N+1) on a disk. (8) Return to the original reading position. (9) Read data for reproduction. (10) When data for reproduction has a divided point, seek at the head of the next continuation field. (11) Resume read-out of data for reproduction. The above operation is repeated.

[0251]In said reference device model, if the following conditions are fulfilled when it postrecords using said reference postrecording algorithm, it can guarantee that there are not overflow of the buffer 504 for postrecording and underflow of the track buffer 502.

[0252]The condition the maximum regeneration time about EU#i which is arbitrary EU in an AV stream $Te(i)$. When maximum recording time of PRU#i which is PRU in $Tr(i)$ and EU#i about the maximum reading time including a division jump is made into $Tw(i)$, it is that <the formula 1> is materialized like the 1st example.

[0253]Since record on a disk of postrecording data is performed synchronizing with the completion of PRU encoding at this time, data in the buffer 504 for postrecording does not accumulate, and there is also no overflow of the buffer 504 for postrecording.

[0254] $Tr(i)$ in <the formula 1> is $Tr(i)=Te(i) \times (Rv+Ra)/Rs+Te(i) \times Rp/Rs+Ta...$ It becomes <the formula 22>.

[0255]The 1st paragraph of the right-hand side and the 2nd paragraph express VU reading time and PRU reading time in EU, respectively. The 3rd paragraph of the right-hand side expresses access time by division jump accompanying read-out. Since division is a maximum of 1 time all over EU, it is the access time of a draft.

[0256] $Tw(i)$ is $Tw(i)=2 \times Ta+Te(i) \times Rp/Rs+Ty...$ It becomes <the formula 23>.

[0257]Here, the 1st paragraph of the right-hand side shows both-way access time to PRU. Using maximum-access-time Ta for access time of a round trip of PRU is based on the same reason as the 1st example.

[0258]The 2nd paragraph of the right-hand side expresses time for recording PRU on a disk. Ty which is the 3rd paragraph of the right-hand side expresses maximum recording time other than postrecording data in an ECC block in which PRU both ends are included, and serves as $Ty=2 \times 32 \text{ KB} /$, and Rs . Since not both ends of PRU are necessarily in agreement with an ECC block boundary, a reason which needs such a paragraph is for recording mostly by a maximum of 2 ECC blocks from size of PRU at the time of PRU record.

[0259]Since he is trying to record PRU continuously on a disk as mentioned above, access under PRU record is not generated. It becomes possible to be able to shorten time accompanying PRU record and to hold down a lower limit of the EU regeneration time low as a result by this.

[0260]When <the formula 22> and the <formula 23> are substituted for <the formula 1> and it solves by $Te(i)$, it is condition $Te(i) \geq (3 \times Ta+Ty) \times (Rs)/(Rs-Rv-Ra-2Rp)$ of $Te(i)$ which can guarantee postrecording... <The formula 24> is obtained.

[0261]That is, the EU regeneration time lower limit $Temin$ in which a postrecording guarantee is possible is $Temin=(3 \times Ta+Ty) \times (Rs)/(Rs-Rv-Ra-2Rp)...$ It becomes <the formula 25>.

[0262]At this time, the upper limit $Temax$ of the EU regeneration time is set up as follows.

$Temax=(3 \times TaRs)/(Rs-Rv-Ra-2Rp)+Tvmax$... the <formula 26> -- here, $Tvmax$ is the maximum regeneration time of VU.

Setting up upper limit is based on a reason explained in the 1st example. If restriction of the above [the EU regeneration time] is filled, immobilization or variable will be available for VU regeneration time in a stream.

[0263]In this example, although the number of times of division in inside of EU (the EU head is also included) is made into a maximum of 1 time, it does not matter as for the arbitrary number of times N . Since continuation region length can be relatively shortened by this, there is an advantage that flexibility of arrangement increases. However, it is necessary to change so that Ta of the 3rd paragraph of the <formula 3> right-hand side may be multiplied by N .

[0264]In this example, although the number of times of division in inside of EU (the EU head is also included) is made into a maximum of 1 time, it may restrict as a head of EU, i.e., PRU, is certainly included once or more to each continuation field which constitutes an AV stream. Or it may restrict to each continuation field as certainly perfect EU is included.

[0265]In immobilization of VU regeneration time within a stream, it may restrict with the minimum number of VU contained to each continuation field. Or when the EU regeneration time is fixed value Te within a stream, continuation region length may be restricted more than $M \times Tex (Rv+2Ra)$. M is one or more integers.

[0266]And although regeneration time of EU is set up in this example, when a position of division is limited to a head of EU, regeneration time of a continuation field will be set up and one variation from which AV stream composition differs compared with the 1st example can also be considered again.

[0267]<Buffer size>, next size of the track buffer 502 required at the time of postrecording are explained. Although a view is the same as that of the 1st example, $Ttmax$ and $Twmmax$ in <the formula 11> from a difference in AV stream composition, respectively $Ttmax=TemaxRp/Rs+Tvmax(Rv+Ra)/Rs$, Becoming $Twmmax=2 \times Ta+Ty+TemaxRp/Rs$, the required track buffer size Lt is $Lt \geq (3 \times TemaxRp+Tvmax(Rv+Ra)+(6Ta+2 \times Ty+Tvmax) \times Rs) \times (Rv+Ra+Rp)/Rs...$ It becomes <the formula 27>.

[0268]Although it assumes moving a pickup to RU of a division jump and the past asynchronously in this example for a reason explained in the 1st example, $Temin$ may be set up on the assumption that movement of a pickup to RU of a division jump and the past is performed synchronously. In this case, < type What is necessary is to remove the 3rd paragraph of the right-hand

side of 22>, and just to think. What is necessary is just to remove a paragraph of Ta in <the formula 11> about track buffer size.

[0269]In this example, although only an ECC block in which PRU is contained as a reference postrecording algorithm is recorded, a reference postrecording algorithm which carries out re-recording of the whole AV stream may be used like the 2nd example. In that case, the 2nd paragraph of the right-hand side of <the formula 23> serves as $Te(i) \times (Rv + Ra + Rp) / Rs$. What is necessary is just to set T_{wmax} in <the formula 11> to $2Ta + Ty + T_{max} \times (Rv + Ra + Rp) / Rs$ about track buffer size.

[0270]<Processing at the time of record>, next processing when recording is directed from a user are explained. Since a flow of processing is the same as the 1st example, it explains using drawing 9. AV streams recorded at this time are bit rate $Rv=5Mbps$ of video, and bit rate $Ra=256k$ bps of an audio, and presuppose that it is a stream corresponding to postrecording of VU regeneration time immobilization. Management information of a file system shall already be read on RAM.

[0271]First, it opts for composition of a stream, or composition of a continuation field (Step 701). When 1 VU is constituted from 15 1GOP, $Rv=5Mbps$, $Ra=256k$ bps, and T_{vmax} about 0.5 second are substituted for <the formula 25> and the <formula 26> for $Rs=20Mbps$, and $Ta=1$ second, and less than the more than range 4.22 second 4.72 second of $Te(i)$ is obtained. It becomes the time of $Te(i)=4.5$ seconds to fulfill this condition in T_{vmax} about 0.5 second, and PRU will be inserted for every nine VU.

[0272]In MPEG-1 audio layer-II, at the time of bit rate 256k bps, the regeneration time Taf of AAU will be 0.024 second, size is set to 768 bytes, and area size of PRU at this time is set to 144384 bytes. In a continuation field, nine VU is made to be contained.

[0273]Continuously recordable free space is looked for for nine VU and one PRU. Specifically, $9 \times T_{vmax} \times (Ra + Rv) + 9 \times T_{vmax} \times Ra$, i.e., continuous free space of 24.8 or more Mbit, is looked for with reference to Space Bitmap on RAM102. If it does not exist, a user is told about the ability to stop recording and not record it (Step 702).

[0274]Next, the audio encoder 117 and the video encoder 118 are started, respectively (Step 703). It confirms whether the above data is stored in a buffer for record by 1 ECC block (32 KB) (Step 704), and while being accumulated, Step 708 is repeated from Step 705.

[0275]If accumulated, empty situations of an ECC block on a disk recorded on the next will be investigated with reference to Space Bitmap on RAM (Step 705). If there is no opening, continuous free space which can record nine VU and PRU(s) will be looked for (Step 707), a pickup will be moved to a head of the free space (Step 708), and data for 1 ECC block in the buffer 111 for record will be recorded on a disk (Step 706).

[0276]On the other hand, if data is not stored in the buffer 111 for record, it confirms whether an end of record is directed (Step 709), and Step 704 will be performed if it is not the end of record.

[0277]The following steps are performed when an end of record is directed. First, about data which is less than 32 KB in a buffer for record, dummy data is added to an end and it is made 32 KB (Step 710). Next, the data is recorded on a disk (Step 711 - Step 714). QuickTime management information and file system management information on RAM102 are recorded on the optical disc 106 (Step 715 - Step 716).

[0278]Operation of the audio encoder 117 and the video encoder 118 which are parallel with the above processing, or the multiplexer 113 is explained. Each encoder sends an encoding result to the multiplexer 113, and a multiplexer stores them in the buffer 114 for multiplexing.

[0279]If data for 1 VU, i.e., 1GOP, and AAU reproduced synchronizing with it are accumulated in the buffer 114 for multiplexing, the multiplexer 113 will send 1-VU data to the buffer 111 for record. If that VU is the $9xi$ -th VU (i is zero or more integers) at this time, PRU with above-mentioned size will be previously sent to the buffer 111 for record.

[0280]Reporting that data for 1 VU has been encoded to the host CPU 101, the host CPU 101 updates QuickTime management information on RAM102 based on a number and size of GOP or AAU which constitute VU.

[0281]<Processing at the time of postrecording>, next processing when postrecording is directed from a user are explained. Since a flow of processing is the same as the 1st example, it explains using drawing 10. Here, QuickTime management information about an AV stream which is already the target of postrecording shall be read into RAM102.

[0282]First, it is reported that it investigates whether the QuickTime movie comprises a stream corresponding to postrecording of only one file, otherwise, cannot postrecord to a user (Step 801). It is because what carried out non-destroying edit of the streams by which this was independently recorded on a disk does not have a guarantee which fulfills conditions for postrecording mentioned above.

[0283]Data for playback is read from a head of PRU on a disk including a postrecording starting position (Step 802). At this time, Step 802 is repeated until it reads data for sufficient regeneration time (Step 803).

[0284]Here, data for sufficient regeneration time means only data volume in which reproduction does not break off, even when an interruption period of data read for reproduction is the maximum. Specifically, it is considered as data volume for 4 seconds supposing a case where division accompanying record (the worst about 3 seconds) of PRU and read-out of AV information is jumped continuously (worst 1 second).

[0285]When PRU is read, an ECC block containing PRU is sent to the buffer 111 for postrecording. In order to manage PRU in the buffer 111 for postrecording at this time, it holds to RAM102 by using a group of reproduction time of onset (relative time from a head of an AV stream) of each PRU in the buffer 111 for postrecording, and an address in inside of the buffer 111 for postrecording as a table.

[0286]Next, the video decoder 116, the audio decoder 115, and the audio encoder 117 are started (Step 804). The audio encoder 117 encodes a sampled voice waveform to AAU, and sends it to the multiplexer 113 with a cycle of AAU. In that case, relative time from a head of an AV stream is added about each AAU.

[0287]The multiplexer 113 stores AAU in PRU in the buffer 111 for postrecording based on time added to AAU. If it finishes storing AAU in PRU to the last, an end of encoding of PRU will be notified to the host CPU 101. Next, it is confirmed whether an end of postrecording is directed from a user (Step 805). Data for reproduction is read like Step 802 until encoding of PRU is completed, if not directed (Step 809).

[0288]When encoding of a certain PRU in the buffer 111 for postrecording is completed (Step 806), It asks for an address on the optical disc 106 which should record the PRU using QuickTime management information, i.e., an address with which the PRU was recorded from the first, from playback time of onset of the PRU currently held on a table on RAM102. The pickup 107 is moved to the address (Step 807), and an ECC block containing the PRU is recorded on the optical disc 107 (Step 808).

[0289]On the other hand, if an end of postrecording is directed, it will wait for the completion of encoding of PRU under present

encoding (Step 810), a pickup will be moved in quest of a recording address of the PRU (Step 811), and PRU will be recorded (Step 812). Finally, QuickTime management information is recorded on a disk (Step 813).

[0290]<Processing at the time of reproduction>, next processing when reproduction is directed from a user are explained. Since a flow of processing is the same as the 1st example, it explains using drawing 11. QuickTime management information about an AV stream which is already the reproductive target here shall be read into RAM102.

[0291]Data for playback is read from a head of PRU on the optical disc 107 including a postrecording starting position (Step 901). At this time, Step 901 is repeated until it reads data for sufficient regeneration time (Step 902).

[0292]Here, data for sufficient regeneration time means only data volume in which reproduction does not break off, even when an interruption period of data read for reproduction is the maximum. Specifically, it is considered as data volume for 1 second supposing a case where division accompanying read-out of AV information is jumped (a maximum of 1 second).

[0293]Next, the video decoder 116 and the audio decoder 115 are started (Step 903). It is confirmed whether an end of reproduction is directed from a user (Step 904). If not directed, AV information for reproduction is read (Step 905). It will end, if an end of reproduction is directed.

[0294]The <7th example>, next the 7th example in this invention are described with drawing 18 thru/or drawing 20. A difference between an example of the 1st thru/or 6 and the 7th example is at a point which does not multiplex a field which stores postrecording data to an AV stream. It differs in that it not only performs record of subaudio information, reproducing a video data and main audio information, but it assumes carrying out independently. It differs in that it is taking into consideration carrying out synchronous reproduction of the BGM data specifically recorded separately to a video data and main audio information by non-destroying edit.

[0295]Composition of an AV stream in <gestalt of AV stream> this example is explained. A stream (AV multiplexed stream) which multiplexed an audio and video, and a stream (audio stream) which comprises only audio information exist in an AV stream. Since composition of AV multiplexed stream is the same as that of a stream corresponding to un-[postrecording] in the 1st above-mentioned example, explanation is omitted here.

[0296]Postrecording audio information is stored in an audio stream, and it comprises AAU of integer pieces. Although AV multiplexed stream and an audio stream are stored in another file, respectively, they may be stored in the same file. When a position of division is limited to a head of VU, a continuation field here becomes equivalent to the above-mentioned RU.

[0297]A deciding method of composition of each continuation field at the time of distributing and arranging a <disk arrangement deciding method>, next an above-mentioned AV stream to two or more continuation fields on a disk is explained. In the 1st, 2nd, and 5th example, in the case of postrecording, to deciding composition of a continuation field that reproduction does not break off in this deciding method. A reproduction algorithm (reference playback algorithm) used as a device (reference device model) used as a standard and a standard is assumed, Even if it plays an audio stream synchronizing with AV multiplexed stream, composition of a continuation field is decided that neither video nor an audio breaks off.

[0298]The reason is explained below. When a field for recording postrecording audio information is multiplexed to an AV stream like the 1st, 2nd, and 5th example at the time of recording, What is necessary is just to read sequentially from a head of AV multiplexed stream, when carrying out synchronous reproduction of a video data, main audio information, and the postrecording audio information.

[0299]When postrecording audio information has not multiplexed in AV multiplexed stream like this example to it, As well as the time of postrecording when carrying out synchronous reproduction of a video data, main audio information, and the postrecording audio information, it is necessary to go back and forth between AV multiplexed stream and postrecording audio information.

[0300]In this example, it also assumes carrying out synchronous reproduction of AV multiplexed stream recorded independently and the audio stream by non-destroying edit, and flexibility of record is high. Therefore, it is severe as conditions for a direction at the time of playback to make neither video nor an audio break off, but to be reproduced from the time of postrecording. Therefore, a continuation field must be decided on the basis of the time of not postrecording but reproduction.

[0301]About a reference device model, since it is the same as that of a thing of the 1st example mentioned above with drawing 7, only a reference playback algorithm is explained using drawing 18. Numbers from (1) to (6) in drawing 18 are equivalent to numbers from (1) to (6) under following explanation.

[0302]The outline of an algorithm is as follows. (1) Read data for reproduction from the AV multiplexed stream 1001. (2) Perform access to the audio stream 1002 at the same time decoding of data with regeneration time equivalent to N VU of the audio stream 1002 is completed. An access position is the part which ended read-out of the audio stream 1002 last time.

[0303](3) Read AAU with regeneration time equivalent to N VU. (4) Return to the original reading position in the AV multiplexed stream 1001. (5) Read data for reproduction. (6) Perform access to the audio stream 1002 at the same time decoding of data with regeneration time equivalent to N VU of the audio stream 1002 is completed. The above operation is repeated.

[0304]In said reference device model, if the following conditions are fulfilled when it postrecords using said reference postrecording algorithm, it can guarantee that there is no underflow of a buffer for reproduction.

[0305]The condition is always being able to read N VU and being able to read audio information equivalent to N VU moreover, while displaying N VU.

[0306]That is, time required to read regeneration time per VU for T_{av} , and read N VU for TrN and T_{ra} need to fill the following formula, when it is considered as time required for read-out of audio information with regeneration time equivalent to N VU.

$$N \times T_{av} \geq TrN + T_{ra} \dots \text{the } \langle \text{formula 29} \rangle \text{ --- } TrN \text{ is explained first.}$$

[0307]In read-out, it is necessary to also take jump time of division into consideration. Here, in a continuation field, N or more VU is made to be contained. A division jump which read-out of regeneration time equivalent to N VU takes to this will be a maximum of 1 time. Here, TrN becomes $TrN = T_a + N \times T_{av} \times (R_a + R_v)$ when a main audio in VU and a maximum bit rate of video are set to R_a and R_v , respectively.

[0308]Next, T_{ra} is explained. Time concerning data read in an audio stream becomes (round trip access time to audio stream) + (audio information reading time) + (access time $\times M$ for a division jump within an audio stream). M shows the number of times of a division jump under audio stream read-out here.

[0309]It becomes $T_{ra} = 2 T_a + N \times T_{av} \times R_p / R_s + T_{ax} \times M$ when the bit rate of audio information in an audio stream is set to R_p . Here, in each continuation field which constitutes an audio stream, if AAU beyond regeneration time equivalent to N VU is made to be contained, it will be set to $M = 1$ and will become $T_{ra} = 3 T_a + N \times T_{av} \times R_p / R_s$.

[0310]The following formula will be obtained, if T_{ra} and TrN are substituted for <the formula 29> and it collects by N.

$N \geq (4 \text{ TaxRs}) / (\text{Tavx} (\text{Rs} - \text{Rv} - \text{Ra} - \text{Rp}))$... It is <formula 30> Got blocked and each continuation field which constitutes AV multiplexed stream needs to consist of VU more than N which fills <the formula 30>, respectively. If this formula is transformed, it will become $N \times \text{Tav} \geq (4 \text{ TaxRs}) / (\text{Rs} - \text{Rv} - \text{Ra} - \text{Rp})$. Since regeneration time T_e of a continuation field is $T_e = N \times \text{Tav}$, it serves as $T_e \geq (4 \text{ TaxRs}) / (\text{Rs} - \text{Rv} - \text{Ra} - \text{Rp})$.

[0311] On the other hand, each continuation field which constitutes an audio stream should just have the regeneration time equivalent to N or more VU in AV multiplexed stream which carries out synchronous reproduction. However, since synchronous reproduction of the audio stream may be carried out combining arbitrary AV multiplexed streams, each continuation field which constitutes an audio stream needs to have regeneration time with a margin.

[0312] The regeneration time T_c specifically needs to fulfill the following conditions, when a maximum bit rate of an audio and video is set to R_{amax} and R_{vmax} .

$T_c \geq (4 \text{ TaxRs}) / (\text{Rs} - \text{Rvmax} - 2R_{\text{amax}})$... Size of the <formula 31> <buffer size>, next the track buffer 502 required at the time of synchronous reproduction of AV multiplexed stream and an audio stream is explained. About size required at the time of postrecording, it sets up based on the 1st example.

[0313] It divides and explains to size L_{t2} [required for a video data, and size L_{t1} / required for main audio information / and subaudio information].

[0314] First, size required for main audio information and subaudio information is explained. It is considered to be a case where a division jump occurs immediately after a division jump occurs once and moreover returns during audio stream read-out at AV multiplexed stream read-out about main audio information and subaudio information that read-out of data breaks off most.

[0315] Therefore, it is necessary to prepare size of only the track buffer 502 which can continue reproduction at worst in the meantime. The size L_{t1} is $L_{t1} \geq (\text{Tra} + \text{Ta} + \text{Tav}) \times (\text{Rv} + \text{Ra})$... It becomes <the formula 32>.

[0316] Next, size required for subaudio information is explained. It is considered to be a case where a division jump occurs immediately after a division jump occurs once and moreover returns during AV multiplexed stream read-out at audio stream read-out about subaudio information that read-out of data breaks off most.

[0317] Therefore, it is necessary to prepare size of only the track buffer 502 which can continue reproduction at worst in the meantime. The size L_{t2} is $L_{t2} \geq (\text{TrN} + \text{Ta} + \text{Tav} + 2 \times \text{Ta}) \times (\text{Rv} + \text{Ra})$... It becomes <the formula 33>.

[0318] <Processing at the time of record>, next processing when recording is directed from a user are explained. Since a flow of processing is the same as that of a thing of drawing 9, explained in the 1st example, only a point of difference is explained. AV multiplexed stream recorded at this time presupposes that they are bit rate $R_v = 5\text{Mbps}$ of video, and bit rate $R_a = 256\text{k bps}$ of an audio. Management information of a file system shall already be read on RAM.

[0319] First, composition of a continuation field is set up (Step 701). Supposing it constitutes 1 VU from 30 2GOP, $R_v = 5\text{Mbps}$, $R_a = 256\text{k bps}$, and $\text{Tav} =$ about 1 second will be substituted for <the formula 30> for $R_s = 20\text{Mbps}$, and $\text{Ta} = 1$ second, and it will be set to $N \geq 5.43$. Then, it is referred to as $N = 6$. That is, in each continuation field, six VU is made to be contained.

[0320] At Step 702 and Step 707, a field which can record six or more VU is looked for. A multiplexer does not perform multiplexing of PRU. About other steps, since it is the same as that of the 1st above-mentioned example, the explanation is omitted.

[0321] <Processing at the time of postrecording>, next processing when postrecording is directed from a user are explained along with drawing 19. QuickTime management information about AV multiplexed stream which is already the target of postrecording shall be read into RAM102.

[0322] Data for reproduction is read from a head of VU of a postrecording starting position (Step 1101). Step 1101 is repeated until it reads data for sufficient regeneration time for data at this time (Step 1102).

[0323] Here, data for sufficient regeneration time means only data volume in which reproduction does not break off, even when an interruption period of data read for reproduction is the maximum. Specifically, it is considered as data volume for 4 seconds supposing a case where division accompanying record (the worst about 3 seconds) of PRU and read-out of AV information is jumped continuously (worst 1 second).

[0324] Next, the video decoder 116, the audio decoder 115, and the audio encoder 118 are started (Step 1103). The audio encoder 118 encodes a sampled voice waveform to AAU, and sends it to a multiplexer with a cycle of AAU. If encoding of audio information of regeneration time (i.e., 6 seconds) equivalent to six VU finishes, an end of encoding will be notified to a host CPU.

[0325] It is confirmed whether an end of postrecording is directed from a user (Step 1104). AV information for reproduction is read until encoding is completed, if not directed (Step 1106).

[0326] If encoding is completed, a pickup will be moved to a recording position of an audio stream, and postrecording data will be recorded (Step 1107). Movement destinations are a head of free space continuous at the time of a postrecording start, and the postrecording data recording end position of last time [it / or subsequent ones].

[0327] When a maximum bit rate of an audio and video is set to 256 [kbps] and 15 [Mbps], respectively, From regeneration time of audio information which should be recorded continuously being 17.9 seconds from <the formula 18>, size of continuous free space is 17.9. [Second] $\times 256$ [kbps] More than 572800 [byte] is secured. When a termination of free space which continued into postrecording data recording is reached, free space where it continued more than 572800 [byte] is looked for, and record is continued from the head.

[0328] If an end of postrecording is directed, it will wait for the completion of encoding of postrecording data under present encoding (Step 1108), the pickup 107 will be moved to a termination of the audio stream, and postrecording data will be recorded (Step 1109).

[0329] As it is finally shown that the above-mentioned audio stream is reproduced synchronizing with the above-mentioned AV multiplexed stream, A track corresponding to the above-mentioned audio stream is added to QuickTime management information of the above-mentioned AV multiplexed stream, and it records on the optical disc 106 (Step 1110).

[0330] <Processing at the time of reproduction>, next processing when reproduction is directed from a user are explained along with drawing 20. QuickTime management information about AV multiplexed stream and an audio stream which are already the reproductive targets shall be read into RAM102.

[0331] First, audio information is read from a position corresponding to a reproduction starting position in an audio stream (Step 1201). Let data volume to read be the regeneration time equivalent to $2 \times N \times \text{Tav}$. That is, it is considered as data volume for 12 seconds here. 1 is set to the variable i.

[0332] Data for reproduction is read from a head of VU of a reproduction starting position (Step 1202). Step 1202 is repeated

until it reads data for sufficient regeneration time at this time (Step 1203).

[0333]Here, data for sufficient regeneration time means only data volume in which reproduction does not break off, even when an interruption period of data read for reproduction is the maximum. Specifically, it is considered as data volume for 5 seconds supposing a case where division accompanying read-out (the worst about 4 seconds) of postrecording data and read-out of AV information is jumped continuously (worst 1 second).

[0334]Next, a video decoder and an audio decoder are started (Step 1204). It is confirmed whether an end of reproduction is directed from a user (Step 1205). If not directed, it is confirmed whether time of $N \times T_{avxi}$ passed from a reproduction start. (Step 1206).

[0335]If it has passed, postrecording data with regeneration time equivalent to N VU will be read. 1 is added to i after that. If it has not passed, VU is read like Step 1202 (Step 1207). It will end, if an end of reproduction is directed.

[0336]The <8th example>, next the 8th example in this invention are described using drawing 21. Here, since the 8th example is similar with the 1st example mentioned above, it is extracted and described in a point of difference. A definition in the 1st example is used for a sign which is not newly defined.

[0337]Composition of an AV stream in <gestalt of AV stream> this example is explained. Unlike the 1st example, in this example, regeneration time of VU is not limited to about 1 second. A stream corresponding to postrecording contains PRU in each VU, as shown in drawing 21.

[0338]Audio information in the same VU and audio information by which synchronous reproduction is carried out are stored in PRU. Here, although PRU is arranged at the head of VU, you may place between audio information and a video data.

[0339]Each VU shall be regeneration time which is different within a stream, respectively. Inside of VU (VU head is also included.) However, division which removes a head of an AV stream may be a maximum of 1 time. On the other hand, a stream corresponding to un-[postrecording] serves as composition excluding PRU from drawing 21.

[0340]About a deciding method of area size of PRU, since it is common, explanation is abbreviated to the 1st example.

[0341]<Disk arrangement deciding method> A deciding method of VU regeneration time in a stream corresponding to un-[postrecording] is explained first. Here, seamless reproduction does not fail in a reference device model of drawing 7, i.e., VU regeneration time is set up so that there may be no underflow of the track buffer 502.

[0342]What is necessary is just to be able to read at least one-piece VU from a decoding start of a certain VU to before a decoding start of following VU also on the worst conditions, in order to guarantee that the track buffer 502 does not carry out underflow.

[0343]Here, the worst conditions are explained. First, a sign is defined. When i -th VU in a stream is made into $VU\#i$, regeneration time of $VU\#i$ is made into $Td(i)$. The minimum VU regeneration time in a stream is set to $Tdmin$, and the greatest VU regeneration time is set to $Tdmax$. Suppose that it has a relation of $Tdmax = Tdmin + d$.

[0344]At this time, the worst conditions are the cases where only $VU(i)$ exists on the track buffer 502 at the time of a read-out start of $VU(i+1)$, when it is $Td(i) = Tdmin$ and $Td(i+1) = Tdmax$. It is because VU which should be read is the largest and time which can moreover be used on read-out is the smallest.

[0345]Since VU certainly exists in the track buffer 502 just before decoding of each VU if read-out of VU can be performed on this condition, the track buffer 502 does not carry out underflow about between [whole] regeneration phases.

[0346]In order to be able to read VU on said worst conditions, it is necessary to fill the following formulas.

$Tdmin \geq T_{rmax} + T_a$... the <formula 34> -- T_{rmax} of the 1st paragraph of the right-hand side shows time concerning read-out of VU of the regeneration time $Tdmax$ here -- $T_{rmax} = Tdmax \times (Rv + Ra) / Rs = (Tdmin + d) \times (Rv + Ra) / Rs$... it becomes <the formula 35>.

[0347]The 2nd paragraph of the right-hand side expresses time concerning a division jump under VU read-out. Here, since division in VU is a maximum of 1 time, it serves as $1 \times T_a$. When <the formula 35> is substituted for <the formula 34> and it collects by $Tdmin$, it is $Tdmin \geq (d \times (Rv + Ra) + T_a \times Rs) / (Rs - Rv - Ra)$... <The formula 36> is obtained.

[0348]That is, according to the bit rate of data, it is necessary to decide the maximum $Tdmax$ and the minimum $Tdmin$ of VU regeneration time in a stream so that the above-mentioned formula may be filled. As long as there will be no disturbance, such as a shock, after that if one VU exists in the track buffer 502 by an initial state when an upper type is filled, seamless reproduction will always be guaranteed.

[0349]How to decide VU regeneration time maximum first and to decide the minimum as how to decide regeneration time based on an upper type can be considered.

[0350]Decoding VU of the regeneration time $Tdmax$ at least, the size of the track buffer 502 needs to read VU of the regeneration time $Tdmax$, and needs to prepare it as much as possible. When the section whose VU in the track buffer 502 has been decoded cannot be reused, the size serves as $2 \times (Rv + 2Ra) \times Tdmax$.

[0351]Next, a deciding method of regeneration time of VU in a stream corresponding to postrecording is explained. Regeneration time is set up so that seamless reproduction may not fail supposing a reference device model and a reference postrecording algorithm, when it postrecords like the 6th example mentioned above using them.

[0352]In said reference device model, if the following conditions are fulfilled when it postrecords using said reference postrecording algorithm, it can guarantee that there are not overflow of the buffer 504 for postrecording and underflow of the track buffer 502.

[0353]Also on the worst conditions, the condition is being able to read at least one-piece VU from a decoding start of a certain VU to before a decoding start of following VU.

[0354]Here, the worst conditions are the cases where only $VU(i)$ exists on the track buffer 502 at the time of a read-out start of $VU(i+1)$, when it is $Td(i) = Tdmin$ and $Td(i-1) = Td(i+1) = Tdmax$. It is because VU which should be read is the largest and time which can moreover be used on read-out is the smallest.

[0355]On this worst condition, since VU certainly exists in the track buffer 502 just before decoding of each VU if read-out of VU can be performed, the track buffer 502 does not carry out underflow about the whole postrecording period.

[0356]Since record on a disk of postrecording data is performed synchronizing with the completion of PRU encoding, data in the buffer 504 for postrecording does not accumulate, and there is also no overflow of the buffer 504 for postrecording.

[0357]On the worst above-mentioned conditions, in order to be able to read VU, it is necessary to fill the following formulas.

$Tdmin \geq T_{rmax} + T_a + T_{wmax}$... the <formula 37> -- here, T_{rmax} of the 1st paragraph of the <formula 37> right-hand side, It is the time which read-out of VU of the regeneration time $Tdmax$ takes, and is $T_{rmax} = Tdmax \times (Rv + Ra + Rp) / Rs = (Tdmin + d) \times (Rv + Ra + Rp) / Rs$... It becomes <the formula 38>.

[0358]The 2nd paragraph of the <formula 37> right-hand side expresses time concerning a division jump under VU read-out.

Here, since division in VU is a maximum of 1 time, it serves as $1 \times Ta$. T_{wmax} of the 3rd paragraph of the <formula 37> right-hand side is time which record of PRU of the regeneration time T_{dmax} takes, and is $T_{wmax} = 3 Ta + T_{dmax} \times R_p / R_s + T_y = 2 Ta + (T_{dmin} + d) \times R_p / R_s + T_y$... It becomes <the formula 39>.

[0359] Here, T_y expresses maximum recording time other than postrecording data in an ECC block in which PRU both ends are included, and serves as $T_y = 2 \times 32 \text{ KB} / R_s$. Three accesses express access of a round trip of PRU, and access under PRU record, respectively. Like the 6th example, if PRU is always recorded on a continuation field, access can be managed with 2 times.

[0360] When <the formula 38> and the <formula 39> are substituted for <the formula 37> and it collects by T_{dmin} , it is $T_{dmin} = (4 Ta + T_y) (x R_s + dx (R_v + R_a + R_p)) / (R_s - R_v - R_a - 2 R_p)$... It becomes <the formula 40>. That is, it is necessary to decide the maximum T_{dmax} and the minimum T_{dmin} of VU regeneration time in a stream so that <the formula 40> may be filled according to the bit rate of data.

[0361] In both sides of a stream corresponding to postrecording, and a stream corresponding to un-[postrecording], this example is available as for the arbitrary number of times N , although the number of times of division in inside of VU is made into a maximum of 1 time. Since continuation region length can be relatively shortened by this, there is an advantage that flexibility of arrangement increases. In that case, it is necessary to change so that T_a of the 2nd paragraph of the <formula 35> right-hand side may be multiplied by N .

[0362] In this example, although the number of times of division within VU (VU head is included) is made into a maximum of 1 time, it may restrict as a head of VU is certainly included once or more to each continuation field which constitutes an AV stream. Or it may restrict to each continuation field as certainly perfect VU is contained.

[0363] When VU regeneration time is the fixed value T_d within a stream, it is continuation region length (1) In the case of a stream corresponding to postrecording, in the case of a stream corresponding to un-[(2) postrecording], more than $T_{dx} (R_v + R_a + R_p)$ may restrict more than $T_{dx} (R_v + R_a)$.

[0364] Decoding VU of the regeneration time T_{dmax} at least, the size of the track buffer 502 needs to read VU of the regeneration time T_{dmax} , and needs to prepare it as much as possible. When the section whose VU in the track buffer 502 has been decoded cannot be reused, the size serves as $2 \times (R_v + R_a + R_p) \times T_{dmax}$.

[0365] Although it assumes moving a pickup to RU of a division jump and the past asynchronously for a reason explained to the 1st example in this example, T_{emin} may be set up on the assumption that movement of a pickup to PRU of a division jump and the past is performed synchronously. In this case, < type What is necessary is to remove the 2nd paragraph of the right-hand side of 37>, and just to think.

[0366] In this example, although only an ECC block in which PRU is contained as a reference postrecording algorithm is recorded, a reference postrecording algorithm which carries out re-recording of the whole AV stream may be used like the 2nd example. In that case, R_p of <the formula 38> will be transposed to $(R_v + R_a + R_p)$.

[0367] If a difference between this example and the 6th example has at least one VU in the track buffer 502 in this example at the time of VU decoding start, and there is no disturbance, such as a shock, the 6th example is at a point which cannot be guaranteed to the ability to guarantee seamless reproduction while after that postrecords. About record, reproduction, and postrecording processing, it is the same as that of the 6th above-mentioned example.

[0368]

[Effect of the Invention] As explained above, according to this invention, the unit which records the inside of an AV stream continuously, control of pickup moving performance, a data transfer rate, the bit rate of data, and the data rewrite of a postrecording field, and ** — it becomes possible by determining by inner one to ensure real-time postrecording.

[Translation done.]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2003-59196
(P2003-59196A)

(43) 公開日 平成15年2月28日 (2003.2.28)

(51) Int.Cl. ⁷	識別記号	F I	データ* (参考)
G 1 1 B 20/12		G 1 1 B 20/12	5 C 0 5 3
	1 0 3		1 0 3 5 D 0 4 4
7/0045		7/0045	Z 5 D 0 9 0
20/10	3 1 1	20/10	3 1 1 5 D 1 1 0
27/00		27/00	D

審査請求 未請求 請求項の数32 O L (全 43 頁) 最終頁に続く

(21) 出願番号 特願2002-5706 (P2002-5706)

(22) 出願日 平成14年1月15日 (2002.1.15)

(31) 優先権主張番号 特願2001-5826 (P2001-5826)

(32) 優先日 平成13年1月15日 (2001.1.15)

(33) 優先権主張国 日本 (J P)

(31) 優先権主張番号 特願2001-170444 (P2001-170444)

(32) 優先日 平成13年6月6日 (2001.6.6)

(33) 優先権主張国 日本 (J P)

(71) 出願人 000005049
シャープ株式会社
大阪府大阪市阿倍野区長池町22番22号

(72) 発明者 木山 次郎
大阪府大阪市阿倍野区長池町22番22号 シャープ株式会社内

(72) 発明者 岩野 裕利
大阪府大阪市阿倍野区長池町22番22号 シャープ株式会社内

(74) 代理人 100103296
弁理士 小池 隆彌 (外1名)

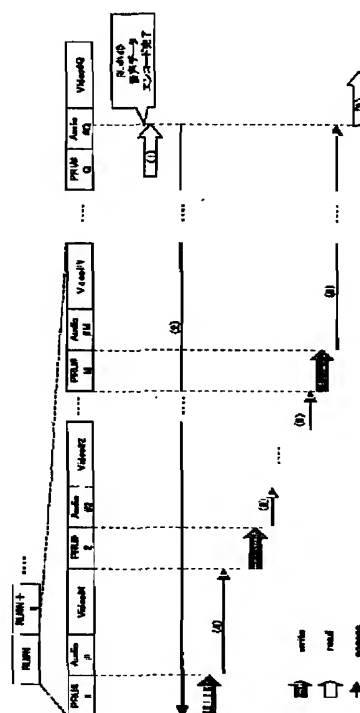
最終頁に続く

(54) 【発明の名称】 データ記録方法及びデータ記録装置並びに記録媒体

(57) 【要約】

【課題】 データ転送速度の比較的低いディスクドライブで、しかもAVストリームがディスク上に分断されて記録されていても、途切れなく再生しながらのアフレコを可能にする。

【解決手段】 映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとを、記録媒体上で連続的に配置して第1のユニットを構成し、記録媒体に記録するデータ記録方法であって、前記第1のユニットの記録単位の大きさを、ピックアップ移動性能、データ転送レート、データのビットレート、前記第1のデータを再生しながらの前記第2のデータ書き換えの制御、のうちのーに基づき決定する。



【特許請求の範囲】

【請求項1】 映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとを、記録媒体上で連続的に配置して第1のユニットを構成し、記録媒体に記録するデータ記録方法であって、前記第1のユニットの記録単位の大きさを、ピックアップ移動性能、データ転送レート、データのビットレート、前記第1のデータを再生しながらの前記第2のデータ書き換えの制御、のうちの一に基づき決定することを特徴とするデータ記録方法。

【請求項2】 前記第1のユニットの記録単位の大きさを決定する際の、上限が設定されることを特徴とする前記請求項1に記載のデータ記録方法。

【請求項3】 前記第1のユニットの記録単位の大きさを決定する際の、下限が設定されることを特徴とする前記請求項1に記載のデータ記録方法。

【請求項4】 映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとを、記録媒体上で連続的に配置して第1のユニットを構成し、記録媒体に記録するデータ記録方法であって、前記第2のデータを書き換える際に用いるメモリ量を、ピックアップ移動性能、データ転送レート、データのビットレート、前記第2のデータ書き換えの制御、のうちの一に基づき決定することを特徴とするデータ記録方法。

【請求項5】 前記第2のデータ書き換えの制御は、前記第1のユニット全体を書き換える制御であることを特徴とする前記請求項1乃至4のいずれかに記載のデータ記録方法。

【請求項6】 前記第2のデータ書き換えの制御は、前記第2のデータのみを書き換える制御であることを特徴とする前記請求項1乃至4のいずれかに記載のデータ記録方法。

【請求項7】 前記第2のデータ書き換えの制御は、前記第2のデータの始端および終端の少なくともいずれかを含む誤り訂正ブロックを一旦読み込んでから行われることを特徴とする前記請求項6に記載のデータ記録方法。

【請求項8】 前記誤り訂正ブロックの読み込みは、前記第1のデータの読み込み時に行われることを特徴とする前記請求項7に記載のデータ記録方法。

【請求項9】 前記第1のユニットは、独立再生可能な1個以上の第2のユニットから構成されることを特徴とする前記請求項1乃至8のいずれかに記載のデータ記録方法。

【請求項10】 映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとを、記録媒体上で連続的に配置して第1のユニットを構成し、記録媒体に記録するデータ記録方法であって、

前記第1のユニットが、独立再生可能な第2のユニットで構成され、

前記第2のユニットには、前記第2のデータを格納する第3のユニットが含まれ、

前記第1のユニットの記録単位の大きさを、ピックアップ移動性能、データ転送レート、データのビットレート、前記第1のデータを再生しながらの前記第2のデータ書き換えの制御、のうちの一に基づき決定し、前記前記第2データ書き換えの制御は、1個以上の第3のユニット毎に書き換える制御であることを特徴とするデータ記録方法。

【請求項11】 映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとを、記録媒体上で連続的に配置して第1のユニットを構成し、記録媒体に記録するデータ記録方法であって、

前記第1のユニットが、独立再生可能な第2のユニットで構成され、

前記第1のデータを再生しながらの前記第2のデータ書き換えの制御が、前記第2のデータのみを書き換える制御であり、

前記第1のユニットの記録単位の大きさを、前記第2のユニットの再生時間に基づき決定することを特徴とするデータ記録方法。

【請求項12】 映像又は音声からなる第1のデータのみ及び前記第1のデータと同期して再生される第2のデータを、記録媒体上で連続的に配置して第1のユニットを構成し、記録媒体に記録するデータ記録方法であって、

前記第2のデータが存在する場合としない場合とによって、前記第1のユニットの記録単位の決定方法を異ならせることを特徴とするデータ記録方法。

【請求項13】 前記第1のユニットの記録単位は、再生時間で規定されることを特徴とする前記請求項1乃至12のいずれかに記載のデータ記録方法。

【請求項14】 映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとを、記録媒体上で連続的に配置して第1のユニットを構成し、記録媒体に記録するデータ記録方法であって、

前記第2のデータを格納するための第1のユニット中の領域を確保するための基準のビットレートを、前記第1のデータのビットレートとは独立に設定することを特徴とするデータ記録方法。

【請求項15】 前記第1のユニット中の領域を確保するための基準のビットレートを、前記第2のデータの最大のビットレートとすることを特徴とする前記請求項14に記載のデータ記録方法。

【請求項16】 前記第1のユニット中の領域を確保するための基準のビットレートを、前記第1のデータ中の

音声のビットレートより低いビットレートとすることを特徴とする前記請求項15に記載のデータ記録方法。

【請求項17】 映像又は音声からなる第1のデータのみ/及び前記第1のデータと同期して再生される第2のデータを、記録媒体上で連続的に配置して第1のユニットを構成し、記録媒体に記録するデータ記録方法であって、

前記第2のデータが存在しない場合には、前記第1のユニットが、記録媒体上で連続的に配置される単位である複数の第2のユニットから構成され、

前記第2のデータが存在する場合には、第1のユニットが、前記第2のユニット単独で構成されることを特徴とするデータ記録方法。

【請求項18】 映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータを、記録媒体上で連続的に配置して第1のユニットを構成し、第1の記録媒体に記録するデータ記録方法であって、

前記第1のデータを再生しながら前記第2のデータを記録する際、一旦第2の記録媒体上の記録領域に記録することを特徴とするデータ記録方法。

【請求項19】 前記第2のデータ記録後、前記第2の記録媒体上の記録領域から、前記第1の記録媒体上の前記第1のユニットに移動することを特徴とする前記請求項18に記載のデータ記録方法。

【請求項20】 前記第1のデータの再生時に前記第1の記録媒体に記録できない前記第2のデータのみ、前記第2の記録媒体に記録することを特徴とする前記請求項18又は19に記載のデータ記録方法。

【請求項21】 前記第2の記録媒体は、前記第1の記録媒体と同一の記録媒体であることを特徴とする前記請求項18乃至20のいずれかに記載のデータ記録方法。

【請求項22】 前記第2の記録媒体上の記録領域は、前記第1のユニット上の領域であることを特徴とする前記請求項21に記載のデータ記録方法。

【請求項23】 前記第2の記録媒体は、半導体メモリであることを特徴とする前記請求項18乃至20のいずれかに記載のデータ記録方法。

【請求項24】 映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータを、記録媒体上で連続的に配置して第1のユニットを構成し、記録媒体に記録するデータ記録装置であって、

前記第1のユニットの再生時間を、ピックアップ移動性能、データ転送レート、データのビットレート、第2のデータ書き換えの制御、のうちのーに基づき決定する手段を備えたことを特徴とするデータ記録装置。

【請求項25】 映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータを、記録媒体上で連続的に配置して第1のユニット

を構成し、第1の記録媒体に記録するデータ記録装置であって、

前記第1のデータを再生しながら前記第2のデータを記録する際、一旦第2の記録媒体上の記録領域に記録する手段を備えたことを特徴とするデータ記録装置。

【請求項26】 映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとが記録される記録媒体であって、

前記第1のデータ中の所定の再生時間分のデータと、該データに同期して再生される第2のデータとを第1のユニットとして管理し、

前記第1のユニットの再生時間は、ピックアップ移動性能、データ転送レート、データのビットレート、第2のデータ書き換えの制御、のうちのーに基づいて決定されることを特徴とする記録媒体。

【請求項27】 映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータを、記録媒体に記録するデータ記録方法であって、前記第1のデータ中の所定の再生時間分のデータと、該データに同期して再生される第2のデータとを第1のユニットとして管理し、

前記第1のユニットの再生時間を、前記第1のユニット中の記録媒体上での物理的不連続点の数に基づいて決定することを特徴とするデータ記録方法。

【請求項28】 前記第2のデータのみを記録媒体上で物理的に連続的に記録されるように制御することを特徴とする前記請求項27に記載のデータ記録方法。

【請求項29】 前記第1のユニット中の第1のデータは、独立して再生可能な単位である第2のユニットの集合から構成されることを特徴とする前記請求項27又は28に記載のデータ記録方法。

【請求項30】 音声からなる第1のデータを、記録媒体に記録するデータ記録方法であって、

前記第1のデータの記録媒体上での連続記録時間を、前記第1のデータと同期再生する可能性のある映像及び音声からなる第2のデータの最大ビットレートに基づいて決定することを特徴とするデータ記録方法。

【請求項31】 映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータを、記録媒体に記録するデータ記録装置であって、前記第1のデータ中の所定の再生時間分のデータと、該データに同期して再生される第2のデータとを第1のユニットとして管理する手段と、

前記第1のユニットの再生時間を、前記第1のユニット中の記録媒体上での物理的不連続点の数に基づいて決定する手段とを備えたことを特徴とするデータ記録装置。

【請求項32】 映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとが記録される記録媒体であって、前記第1のデータ中の所定の再生時間分のデータと、該

データに同期して再生される第2のデータとを第1のユニットとして管理し、
前記第1のユニットの再生時間は、前記第1のユニット中の記録媒体上での物理的不連続点の数に基づくことを特徴とする記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、映像データ、音声データをハードディスク、光ディスク等のランダムアクセス可能な記録媒体に対して記録・再生するデータ記録方法及びデータ記録装置に関するものである。

【0002】

【従来の技術】ディスクメディアを用いたビデオや音声のデジタル記録再生装置が普及しつつある。それらにおいて、テープメディアと同様にアフターレコーディング（アフレコ）機能を安価に実現する技術が求められている。アフレコ機能は、既に記録したオーディオやビデオに対し、後から情報、特にオーディオを追記する機能である。

【0003】ディスクメディアを用いてアフレコ機能を実現している従来技術として、例えば特開平5-234084号公報に記載のディスク記録再生装置が知られている。

【0004】この技術は、プログラム再生期間よりデータの読込期間が短いことを利用して、現在再生しているディスクからメモリにデータを読み込んでから次のデータを読み込むまでの間に、入力されたアフレコ音声データをディスクに書き込むというもので、ディスク記録再生手段が1つであってもアフレコを実現することが可能である。

【0005】ここで、プログラム再生期間とは、ビデオや音楽などプログラムそれぞれが持つ固有の再生期間のことである。例えば1分間のビデオは、再生手段が変わったとしても1分間で再生されなければ正確に再生されたとは言えない。

【0006】従来技術におけるディスクの記録フォーマットを図22に示す。ディスクはECC（エラー・コレクション・コーディング）ブロックの列で構成される。ECCブロックは符号化を行う際の最小単位であり、データに加えエラー補正用のパリティが付加され、符号化が行われている。

【0007】データを読み込む際は、この単位で読み込み誤り訂正をしてから、必要なデータを取り出す。一方、データを書き換える際は、まずECCブロック単位で読み込み、誤り訂正をしたデータに対し、必要な部分を書き換え、再度誤り符号の付与を行ない、ディスクに記録を行なう。このことは、1バイト書き換える場合でも、そのバイトが含まれるECCブロック全体を読み込み書き込む必要があることを意味する。

【0008】ビデオやオーディオは、ECCブロック中

で、図22(b)に示すように、アフレコオーディオブロック、オリジナルオーディオブロック、オリジナルビデオブロックの順に配置される。

【0009】それぞれのブロックには、ほぼ同じ時間に対応するアフレコオーディオ、オリジナルオーディオ、オリジナルビデオが含まれている。尚、オリジナルオーディオブロックとオリジナルビデオブロックとを合わせてオリジナルブロックと呼ぶことにする。

【0010】オリジナルプログラム（アフレコオーディオを記録する前の映像）を記録する際は、アフレコオーディオブロックにダミーのデータを書き込んでおく。

【0011】次に、従来技術におけるアフレコ時の動作について、図23とともに説明する。ここで、図23(a)のグラフは、ディスクからの読込、再生や記録といった各処理の時間的な関係を示しており、矢印内の記号は図23(b)のグラフにおける縦軸に対応し、処理対象となっているデータのディスク上での位置を表す。図23(b)はディスク中でのヘッドの位置を、図17(c)のグラフはバッファメモリに占めるプログラムデータの割合を模式的に示している。

【0012】ここではプログラムが、ディスク中のs11～s18への連続的な領域に配置され、s11～s13、s13～s15、s15～s17の各領域がそれぞれECCブロックに対応し、s11～s12、s13～s14、s15～s16、s17～s18の各領域がそれぞれアフレコオーディオブロックに対応しているとする。

【0013】時刻t1の時点ですでにs13までの領域がバッファメモリに格納されており、s11～s13の領域に記録されていたデータがデコードされ再生されるとともに、そのデータのアフレコ音声の入力、エンコードが行われている。

【0014】時刻t1～t3において、領域s13～s15のデータをディスクから読み込み、バッファメモリ及びアフレコバッファへの格納を行う。アフレコバッファは読み込んだECCブロックをそのまま記憶し、図22(b)と同様の構成をとる。時刻t2は、時刻t1の時点で行われていた領域s11～s13に記録されていたデータのデコード、再生が終了する時刻である。

【0015】時刻t2以降は、時刻t1～t3で読み込まれる領域s13～s15のデータをデコード、再生するとともに、そのデータのアフレコ音声の入力、エンコードが行われる。この領域s13～s15のデータのデコード、再生は時刻t5まで行われる。

【0016】時刻t2までに入力されたアフレコ音声は、少なくとも時刻t3までにエンコードが終了する。時刻t3において、時刻t2までに入力されたアフレコ音声をディスク媒体に記録する。このときに、s11にアクセスする際、ディスクの回転待ちの時間を要するが、ディスクの読み書きの時間に比べると、短時間であるので、ここでは考慮しない。

【0017】アフレコ音声のディスクへの書き込みは、時刻t3～t4で行われる。このディスクへの書き込みが時刻t4で終了すると、時刻t4から領域s15～s17のデータをディスクから読み込む。このように、以下同様の処理を繰り返す。

【0018】上述の従来技術では、情報圧縮を行うことにより、データの再生時間よりも読み込み時間が短くなることを利用し、記録再生手段を、記録と再生で時分割して利用することで、1つの記録再生手段だけでアフレコを実現している。尚、特開2001-118362号公報にも、同様の技術が開示されている。

【0019】

【発明が解決しようとする課題】しかしながら、上述した従来のディスク記録再生装置においては、ディスクのデータ転送速度に比べてAVストリームのビットレートが十分低い場合は良いが、データ入出力速度の余裕が小さい場合は、アフレコを行いながら、途切れなく再生することは困難である。

【0020】本発明は、上記課題に鑑みてなされたものであり、データ転送速度の比較的低いディスクドライブで、しかもAVストリームがディスク上に分断されて記録されていても、途切れなく再生しながらのアフレコを可能にすることを目的とする。

【0021】

【課題を解決するための手段】本願の第1の発明は、映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとを、記録媒体上で連続的に配置して第1のユニットを構成し、記録媒体に記録するデータ記録方法であって、前記第1のユニットの記録単位の大きさを、ピックアップ移動性能、データ転送レート、データのビットレート、前記第1のデータを再生しながらの前記第2のデータ書き換えの制御、のうちのーに基づき決定することを特徴とする。

【0022】本願の第2の発明は、前記第1のユニットの記録単位の大きさを決定する際の、上限が設定されることを特徴とする。

【0023】本願の第3の発明は、前記第1のユニットの記録単位の大きさを決定する際の、下限が設定されることを特徴とする。

【0024】本願の第4の発明は、映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとを、記録媒体上で連続的に配置して第1のユニットを構成し、記録媒体に記録するデータ記録方法であって、前記第2のデータを書き換える際に用いるメモリ量を、ピックアップ移動性能、データ転送レート、データのビットレート、前記第2のデータ書き換えの制御、のうちのーに基づき決定することを特徴とする。

【0025】本願の第5の発明は、前記第2のデータ書き換えの制御が、前記第1のユニット全体を書き換える

制御であることを特徴とする。

【0026】本願の第6の発明は、前記第2のデータ書き換えの制御が、前記第2のデータのみを書き換える制御であることを特徴とする。

【0027】本願の第7の発明は、前記第2のデータ書き換えの制御が、前記第2のデータの始端および終端の少なくともいずれかを含む誤り訂正ブロックを一旦読み込んでから行われることを特徴とする。

【0028】本願の第8の発明は、前記誤り訂正ブロックの読み込みが、前記第1のデータの読み込み時に行われることを特徴とする。

【0029】本願の第9の発明は、前記第1のユニットが、独立再生可能な1個以上の第2のユニットから構成されることを特徴とする。

【0030】本願の第10の発明は、映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとを、記録媒体上で連続的に配置して第1のユニットを構成し、記録媒体に記録するデータ記録方法であって、前記第1のユニットが、独立再生可能な第2のユニットで構成され、前記第2のユニットには、前記第2のデータを格納する第3のユニットが含まれ、前記第1のユニットの記録単位の大きさを、ピックアップ移動性能、データ転送レート、データのビットレート、前記第1のデータを再生しながらの前記第2のデータ書き換えの制御、のうちのーに基づき決定し、前記前記第2データ書き換えの制御は、1個以上の第3のユニット毎に書き換える制御であることを特徴とする。

【0031】本願の第11の発明は、映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとを、記録媒体上で連続的に配置して第1のユニットを構成し、記録媒体に記録するデータ記録方法であって、前記第1のユニットが、独立再生可能な第2のユニットで構成され、前記第1のデータを再生しながらの前記第2のデータ書き換えの制御が、前記第2のデータのみを書き換える制御であり、前記第1のユニットの記録単位の大きさを、前記第2のユニットの再生時間に基づき決定することを特徴とする。

【0032】本願の第12の発明は、映像又は音声からなる第1のデータのみ及び前記第1のデータと同期して再生される第2のデータを、記録媒体上で連続的に配置して第1のユニットを構成し、記録媒体に記録するデータ記録方法であって、前記第2のデータが存在する場合としない場合とによって、前記第1のユニットの記録単位の決定方法を異ならせることを特徴とする。

【0033】本願の第13の発明は、前記第1のユニットの記録単位が、再生時間で規定されることを特徴とする。

【0034】本願の第14の発明は、映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとを、記録媒体上で連続的に配置し

て第1のユニットを構成し、記録媒体に記録するデータ記録方法であって、前記第2のデータを格納するための第1のユニット中の領域を確保するための基準のビットレートを、前記第1のデータのビットレートとは独立に設定することを特徴とする。

【0035】本願の第15の発明は、前記第1のユニット中の領域を確保するための基準のビットレートを、前記第2のデータの最大のビットレートとすることを特徴とする。

【0036】本願の第16の発明は、前記第1のユニット中の領域を確保するための基準のビットレートを、前記第1のデータ中の音声のビットレートより低いビットレートとすることを特徴とする。

【0037】本願の第17の発明は、映像又は音声からなる第1のデータのみ/及び前記第1のデータと同期して再生される第2のデータを、記録媒体上で連続的に配置して第1のユニットを構成し、記録媒体に記録するデータ記録方法であって、前記第2のデータが存在しない場合には、前記第1のユニットが、記録媒体上で連続的に配置される単位である複数の第2のユニットから構成され、前記第2のデータが存在する場合には、第1のユニットが、前記第2のユニット単独で構成されることを特徴とする。

【0038】本願の第18の発明は、映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとを、記録媒体上で連続的に配置して第1のユニットを構成し、第1の記録媒体に記録するデータ記録方法であって、前記第1のデータを再生しながら前記第2のデータを記録する際、一旦第2の記録媒体上の記録領域に記録することを特徴とする。

【0039】本願の第19の発明は、前記第2のデータ記録後、前記第2の記録媒体上の記録領域から、前記第1の記録媒体上の前記第1のユニットに移動することを特徴とする。

【0040】本願の第20の発明は、前記第1のデータの再生時に前記第1の記録媒体に記録できない前記第2のデータのみ、前記第2の記録媒体に記録することを特徴とする。

【0041】本願の第21の発明は、前記第2の記録媒体が、前記第1の記録媒体と同一の記録媒体であることを特徴とする。

【0042】本願の第22の発明は、前記第2の記録媒体上の記録領域が、前記第1のユニット上の領域であることを特徴とする。

【0043】本願の第23の発明は、前記第2の記録媒体が、半導体メモリであることを特徴とする。

【0044】本願の第24の発明は、映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとを、記録媒体上で連続的に配置して第1のユニットを構成し、記録媒体に記録するデータ

記録装置であって、前記第1のユニットの再生時間を、ピックアップ移動性能、データ転送レート、データのビットレート、第2のデータ書き換えの制御、のうちのーに基づき決定する手段を備えたことを特徴とする。

【0045】本願の第25の発明は、映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとを、記録媒体上で連続的に配置して第1のユニットを構成し、第1の記録媒体に記録するデータ記録装置であって、前記第1のデータを再生しながら前記第2のデータを記録する際、一旦第2の記録媒体上の記録領域に記録する手段を備えたことを特徴とする。

【0046】本願の第26の発明は、映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとが記録される記録媒体であって、前記第1のデータ中の所定の再生時間分のデータと、該データに同期して再生される第2のデータとを第1のユニットとして管理し、前記第1のユニットの再生時間は、ピックアップ移動性能、データ転送レート、データのビットレート、第2のデータ書き換えの制御、のうちのーに基づいて決定されることを特徴とする。

【0047】本願の第27の発明は、映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとを、記録媒体に記録するデータ記録方法であって、前記第1のデータ中の所定の再生時間分のデータと、該データに同期して再生される第2のデータとを第1のユニットとして管理し、前記第1のユニットの再生時間を、前記第1のユニット中の記録媒体上での物理的不連続点の数に基づいて決定することを特徴とする。

【0048】本願の第28の発明は、前記第2のデータのみを記録媒体上で物理的に連続的に記録されるように制御することを特徴とする。

【0049】本願の第29の発明は、前記第1のユニット中の第1のデータが、独立して再生可能な単位である第2のユニットの集合から構成されることを特徴とする。

【0050】本願の第30の発明は、音声からなる第1のデータを、記録媒体に記録するデータ記録方法であって、前記第1のデータの記録媒体上での連続記録時間を、前記第1のデータと同期再生する可能性のある映像及び音声からなる第2のデータの最大ビットレートに基づいて決定することを特徴とする。

【0051】本願の第31の発明は、映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとを、記録媒体に記録するデータ記録装置であって、前記第1のデータ中の所定の再生時間分のデータと、該データに同期して再生される第2のデータとを第1のユニットとして管理する手段と、前記第1のユニットの再生時間を、前記第1のユニット中の記

録媒体上での物理的不連続点の数に基づいて決定する手段とを備えたことを特徴とする。

【0052】本願の第32の発明は、映像又は音声からなる第1のデータと、前記第1のデータと同期して再生される第2のデータとが記録される記録媒体であって、前記第1のデータ中の所定の再生時間分のデータと、該データに同期して再生される第2のデータとを第1のユニットとして管理し、前記第1のユニットの再生時間は、前記第1のユニット中の記録媒体上での物理的不連続点の数に基づくことを特徴とする。

【0053】

【発明の実施の形態】以下、本発明の実施形態について、図面を参照しながら詳細に説明する。

【0054】＜システム構成＞図1は本実施形態において共通に用いる、アフレコ可能なビデオディスクレコーダの概略構成を示すブロック図である。図1に示すように、この装置は、バス100、ホストCPU101、RAM102、ROM103、ユーザインタフェース104、システムクロック105、光ディスク106、ピックアップ107、ECCデコーダ108、ECCエンコーダ109、再生用バッファ110、記録/アフレコ用バッファ111、出マルチプレクサ112、マルチプレクサ113、多重化用バッファ114、オーディオデコーダ115、ビデオデコーダ116、オーディオエンコーダ117、ビデオエンコーダ118、及び図示しないカメラ、マイク、スピーカ、ディスプレイ等を備えている。

【0055】ホストCPU101は、バス100を通じてデマルチプレクサ112、マルチプレクサ113、ピックアップ107、また図示はしていないが、オーディオデコーダ115、ビデオデコーダ116、オーディオエンコーダ117、ビデオエンコーダ118との通信を行う。

【0056】再生時において、光ディスク106からピックアップ107を通じて読み出されたデータは、ECCデコーダ108によって誤り訂正され、再生用バッファ110に一旦蓄えられる。デマルチプレクサ112はオーディオデコーダ115、ビデオデコーダ116からのデータ送信要求に従って、再生用バッファ中のデータをその種別によって適当なデコーダに振り分ける。

【0057】一方、記録時においては、オーディオエンコーダ117とビデオエンコーダ118とによって圧縮符号化されたデータが多重化用バッファ114に一旦送られ、マルチプレクサ113によってAV多重化され、記録/アフレコ用バッファ111に送られる。記録/アフレコ用バッファ111中のデータは、ECCエンコーダ109によって誤り訂正符号が付加され、ピックアップ107を通じて光ディスク106に記録される。

【0058】尚、オーディオデータの符号化方式にはMP EG-1 Layer-IIを、ビデオデータの符号化方式にはMPEG-2をそれぞれ用いる。

【0059】光ディスク106は、外周から内周に向かって螺旋状に記録再生が行われる脱着可能な光ディスクと

する。2048byteを1セクタとし、誤り訂正のため16セクタでECCブロックを構成する。ECCブロック中のデータを書き換える場合、そのデータが含まれるECCブロック全体を読み込み、誤り訂正を行い、対象のデータを書き換えた後、再び誤り訂正符号を付加し、ECCブロックを構成して記録媒体に記録する必要がある。

【0060】また、ここでの光ディスク106は、記録効率を上げるためZCAV(ゾーン角速度一定)を採用しており、記録領域は回転数の異なる複数のゾーンで構成される。

【0061】＜ファイルシステム＞次に、本実施形態においては、光ディスク106上の各種情報を管理するためにファイルシステムを用いる。ファイルシステムには、PCとの相互運用を考慮してUDF(Universal Disk Format)を使用する。ファイルシステム上では各種管理情報やAVストリームはファイルとして扱われる。

【0062】ユーザエリアは2048byteの論理ブロック(セクタと一対一対応)で管理される。各ファイルはディスク上で物理的に連続した論理ブロックで構成される必要は無く、論理ブロック単位で分散して記録されてもよい。また、空き領域はSpace Bitmapを用いて論理ブロック単位で管理される。

【0063】＜ファイルフォーマット＞また、本実施形態では、AVストリーム管理のためのフォーマットとしてQuickTimeファイルフォーマットを用いる。QuickTimeファイルフォーマットとは、Apple社の開発したマルチメディアデータ管理用フォーマットであり、パーソナルコンピュータ(PC)の世界では広く用いられている。

【0064】QuickTimeファイルフォーマットは管理情報とAVストリームとで構成される。ここでは、両者を合わせてQuickTimeムービーと呼ぶ。両者は同じファイル中に存在しても、別々のファイルに存在しても良い。同じファイル中に存在する場合は、図2(a)に示すような構成を取る。各種情報はatomという共通の構造に格納される。管理情報はMovie atomという構造に格納され、AVストリームはMovie data atomという構造に格納される。

【0065】尚、Movie atom中の管理情報には、AVストリーム中の任意の時間に対応するAVデータのファイル中での相対位置を導くためのテーブルや、オーディオデータやビデオデータの属性情報や、後述する外部参照情報等が含まれている。

【0066】一方、管理情報とAVストリームを別々のファイルに格納した場合は、図2(b)に示すような構成を取る。管理情報はMovie atomという構造に格納されるが、AVストリームはatomには格納される必要はない。このとき、Movie atomはAVストリームを格納したファイルを「外部参照」している、という。

【0067】外部参照は、図2(c)に示すように、複数のAVストリームファイルに対して行うことが可能であ

り、この仕組みにより、AVストリーム自体を物理的に移動することなく見かけ上編集を行ったように見せる、いわゆる「ノンリニア編集」「非破壊編集」が可能になる。

【0068】Movie atomの構成を図3(a)に示す。それぞれのatomは特定のatomを内包する構成となっている。Movie atomは、そのMovie atomが管理するプログラムの全体的な属性を管理するMovie header atomやそのプログラムに含まれる各トラックに関する情報を格納する1個以上のTrack atom等を含む。

【0069】それぞれのatomには各種の情報を格納するatomを含むが、ここでは本発明の理解に必要なものに絞って説明する。Movie atomには、User data atomと呼ばれる、QuickTimeフォーマットで定義されてない独自の情報を管理するためのatomを格納可能である。

【0070】本発明では、User data atom中に、図3(b)に示すように、AVストリームの構造(後述するRecord UnitやVideo Unitの構成)に関する情報を格納するrecord-unit description atom、そのプログラムを再生するのに必要な機器の性能に関する情報(シーク時間やディスク転送速度等)を格納するset performance atomを管理するX descriptor atomを追加定義している。

【0071】＜第1実施例＞次に、本発明における第1の実施例について、図4乃至図11とともに説明する。

【0072】＜AVストリームの形態＞まず、本実施例におけるAVストリームの構成について、図4から図6を用いて説明する。図4に示すように、AVストリームは整数個のRecord Unit(RU)で構成される。RUはディスク上で連続的に記録する単位である。

【0073】RUの長さは、AVストリームを構成するRUをどのようにディスク上に配置してもシームレス再生(再生中に絵や音が途切れないで再生できること)やリアルタイムアフレコ(アフレコ対象のビデオをシームレス再生しながらオーディオを記録すること)が保証されるように設定される。

【0074】この設定方法については後述する。また、RU境界がECCブロック境界と一致するようにストリームを構成する。RUのこれらの性質によって、AVストリームをディスクに記録した後も、シームレス再生を保証したまま、ディスク上でRU単位の配置を容易に変更できる。

【0075】RUは、整数個のVideo Unit(VU)で構成する。VUは単独再生可能な単位であり、そのことから再生の際のエントリ・ポイントとなりうる。VUの構成は、アフレコに対応したストリーム(アフレコ対応ストリーム)とアフレコには対応しないストリーム(アフレコ非対応ストリーム)で異なる。

【0076】まず、アフレコ非対応ストリームにおけるVU構成を図5に示す。VUは、1秒程度のビデオデータを格納した整数個のGOP(グループ・オブ・ピクチャ)とそれらと同じ時間に再生されるメインオーディオデータを

格納した整数個のAAU(オーディオ・アクセス・ユニット)で構成される。尚、GOPは、MPEG-2ビデオ規格における圧縮の単位であり、複数のビデオフレーム(典型的には15フレーム程度)で構成される。

【0077】AAUはMPEG-1 LayerII規格における圧縮の単位で、1152点の音波形サンプル点により構成される。サンプリング周波数が48kHzの場合、AAUあたりの再生時間は0.024秒となる。VU中ではAV同期再生のために必要となる遅延を小さくするためAAU、GOPの順に配置する。

【0078】尚、VU中の先頭ビデオフレームの再生開始タイミングは、先頭AAUの再生開始タイミング以前でなければならず、その時間差は1AAUの再生時間未満でなければならない。VUは、シームレス再生のために所定の単位以上で、ディスク上に連続的に記録する必要がある。その単位については後述する。

【0079】また、VU単位で独立再生可能なようにVU中のビデオデータの先頭にはSequenceHeader(SH)を置く。また、後続するVUとビデオエンコードの属性(例えば画面を構成するピクセル数)が変わる場合には、末尾にSequence End Code(SEC)を置く。

【0080】VUの再生時間は、VUに含まれるビデオフレーム数にビデオフレーム周期をかけたものと定義する。また、VUを整数個組み合わせるRUを構成する場合、RUの始末端をECCブロック境界に合わせるため、VUの末尾を0で埋める。

【0081】一方、アフレコ対応ストリームにおけるVUの構成は、図6に示すとおりである。アフレコ非対応ストリームにおけるVUの先頭に、ビデオおよびメインオーディオデータと同時に再生を行うアフレコ(サブオーディオ)データを格納するための領域としてPost Recording Unit(PRU)を設けている。尚、ここではPRUをメインオーディオを格納するための領域の前に置いているが逆でも構わない。

【0082】PRUの領域サイズは、メインオーディオのビットレートに関わらず、1種類あるいは限定された種類の中から選ぶようにする。なぜなら、ビットレートを自由に設定可能とすると、アフレコ機能を持った機器はあらゆるビットレートでのオーディオエンコードをサポートする必要があるためである。

【0083】例えば、PRUの領域サイズは、メインオーディオのビットレートに関わらず、再生可能な最大のオーディオビットレートに基づき確保する。例えばメインオーディオのビットレートが128kbpsであったとしても、同一のVUに含まれるPRUの領域サイズは再生可能最大ビットレート(例えば256kbps)で確保する。

【0084】この場合、アフレコを別の機器で行う場合に、元々のAVストリームを記録した機器のオーディオのビットレートに縛られず、その機器のサポートしたビットレートでアフレコオーディオを記録でき、アフレコを実施する機器におけるエンコード対応の負担が減少す

る。

【0085】一方、PRUの領域サイズをメインオーディオのビットレートに関わらず、低いオーディオビットレートに基づき確保する、という方法もある。このことによって、アフレコを普段行わないが、極たまに必要性が発生する、という多くのユーザにとって、ディスクの記録容量をわずかながらでも節約することが可能となる。アフレコ入力人間の音声であることが多いため、低いビットレートであっても、たいいていの場合音質的に問題ない。

【0086】尚、ここではアフレコの対象はビデオとメインオーディオ、アフレコにおいて記録するデータはサブオーディオデータとしているが、以下の説明は特にそれに限定されるものではない。

【0087】＜AVストリーム管理方法＞AVストリーム上での各RUや各VUの位置や再生時間に関しては、前述のMovie Atomで管理を行う。詳細については、ここでの説明に不要であるため省略する。

【0088】＜ディスク配置決定方法＞まず、アフレコ対応ストリームにおけるRU再生時間の決定方法について、説明する。この決定方法では、機器間での互換性確保のため、基準となるデバイス(リファレンス・デバイス・モデル)と基準となるアフレコアルゴリズム(リファレンス・アフレコ・アルゴリズム)を想定し、次にそれらを用いてアフレコを行った際にシームレス再生が破綻しないようにRU再生時間を決める。

【0089】それではまず、リファレンス・デバイス・モデルについて、図7を用いて説明する。リファレンス・デバイス・モデルは1個のピックアップとそれにつながるECCエンコーダ・デコーダ501、トラックバッファ502、デマルチプレクサ503、アフレコ用バッファ504、オーディオエンコーダ509、ビデオバッファ505、オーディオバッファ506、ビデオデコーダ507、オーディオデコーダ508とによって構成される。

【0090】本モデルでは、ピックアップが1個であるため、再生用データのディスクからの読み出しとアフレコ用データのディスクへの記録は時分割で行う。ディスクから再生用データを読み出す際、PRUも含めて読み出す。読み出されたPRUを含むECCブロック(PRUブロック)は、トラックバッファ502からアフレコ用バッファ504に送られる。

【0091】オーディオエンコーダ509は、AAU周期でアフレコ用バッファ504に出力する。この出力によって、アフレコ用バッファ504中の対応するPRUブロックを上書きする。アフレコデータの記録は、PRUブロックを所定のECCブロックに記録することで行う。

【0092】本モデルにおいて、PRUブロックをトラックバッファ502からアフレコ用バッファ504に送ることを想定しているのは、次の理由による。本実施例におけるAVストリームでは、PRU境界とECCブロック境界とは一致

しないため、PRU境界を含むECCブロックにはPRUのデータだけではなく、その他のデータ(直前のVUのビデオデータや同一のVUのオーディオデータ)が含まれる。

【0093】従って、PRUにデータを記録する際には、PRU境界を含むECCブロックをメモリに一旦読み出す必要がある。PRUを記録する直前にメモリに読み出すということも考えられるが、再生用データ読み出し時にPRU境界を含むECCブロックを必ず読み出していることから、再生用データ読み出し時に読み出したPRUを含むECCブロックを一時的にアフレコ用バッファ504に保持しておくことで、PRU境界を含むECCブロックの再度読み出しを省略している。

【0094】本モデルにおけるシームレス再生は、VUのデコード開始時にトラックバッファ502上に少なくとも1個VUが存在すれば保証されるものとする。オーディオフレームデータのECCエンコーダ501へのデータの入力速度およびECCデコーダ501からデータの出力速度はRsとする。

【0095】また、アクセスによる読み出し、記録の停止する最大期間をTaとする。さらに、短いアクセス(100トラック程度)に要する時間をTkとする。なお、これら期間には、シーク時間、回転待ち時間、アクセス後に最初にディスクから読み出したデータがECCから出力されるまでの時間が含まれる。本実施例では、Rs=20Mbps、Ta=1秒、Tk=0.2秒とする。

【0096】次に、リファレンス・アフレコ・アルゴリズムについて、図8を用いて説明する。尚、図8中の(1)から(9)までの番号は、以下の説明中の(1)から(9)までの番号に対応する。アルゴリズムの概要は次の通りである。

【0097】(1) 再生用データの読み出しを行う。(2) N番目のRUであるRU#Nに対応するオーディオデータのエンコードが終了すると同時に、RU#Nへのピックアップ移動を行う。(3) RU#Nの先頭のPRUであるPRU#1に対応するPRUブロックを記録する。

【0098】(4) RU#N中の2番目のPRUであるPRU#2へピックアップを移動する。(5) PRU#2に対応するPRUブロックを記録する。(6) 次のPRUへのピックアップ移動、PRUブロック記録を繰り返す。(7) RU#N中の最後のPRUであるPRU#Mに対応するPRUブロックを記録する。(8) 元の読み出し位置にピックアップを戻す。(9) 再生用データの読み出しを再開する。以上の動作を繰り返す。

【0099】前記リファレンス・デバイス・モデルにおいて、前記リファレンス・アフレコ・アルゴリズムを用いてアフレコを行った場合、次のような条件を満たせば、トラックバッファ502のアンダーフローがないことが保証できる。

【0100】その条件とは、AVストリーム中の任意のRUであるRU#iについて最大再生時間をT(i)、分断ジャンプを含めた最大読み出し時間をTr(i)、RU#i中のPRUの最大

記録時間を $T_w(i)$ としたとき、
 $T_e(i) \geq T_r(i) + T_w(i) \cdots$ <式 1>
 が成立することである。

【0101】なぜなら、この式は、シームレス再生の十

$$\sum_{i=1}^n T_e(i) \geq \sum_{i=1}^n (T_r(i) + T_w(i)) \cdots$$
 <式 2>

【0103】を満たす十分条件であるためである。

【0104】また、PRUエンコード完了に同期してアフレコデータのディスクへの記録を行っているため、アフレコ用バッファ504中のデータが累積していくことはなく、アフレコ用バッファ504のオーバーフローもない。

【0105】<式 1>中の $T_r(i)$ は、AVストリーム中のオーディオとビデオおよびPRU領域サイズ確保の基準と

$$T_w(i) = 2T_a + (M-1) \times T_k + T_e(i) \times R_p / R_s + (2M-1) \times L_y / R_s \cdots$$
 <式 4>

となる。

【0107】ここで、右辺第1項は、RUへの往復アクセス時間を示す。PRUへの往復のアクセス時間に最大アクセス時間 T_a を用いているのは、以下の理由に基づく。

【0108】現在読み出しているトラックと記録すべきPRUの存在するトラックの距離は、そのときの再生用バッファによる遅延時間に依存する。しかし、遅延時間は再生用バッファサイズによって異なり、また同じバッファサイズであっても、直前に衝撃によって読み出しが一時的に停止した場合にも異なる。すなわちアクセスする距離は不定であり、そのため最悪値で見積もる必要がある。

【0109】右辺第2項は、PRU間をジャンプする時間の合計である。尚、 M はRU# i を構成するVUの個数である。右辺第3項は、RU# i に含まれるPRUをディスクに記録するための時間の合計を表す。右辺第4項は、PRU両端が含まれるECCブロック中のアフレコデータ以外の記録時間の最大値を表している。

【0110】ここで、 L_y はECCブロックサイズである32K

$$T_e(i) \geq 3T_a / (R_s - R_v - R_a - R_p - (T_k / T_v) \times R_s - 2L_y / T_v) \cdots$$
 <式 5>

が得られる。

【0114】つまり、アフレコ保証可能なRU再生時間下

$$T_{min} = 3T_a / (R_s - R_v - R_a - 2R_p - (T_k / T_{vmin}) \times R_s - 2L_y / T_v) \cdots$$
 <式 6>

となる。

【0115】このとき、RU再生時間の上限值 T_{max} を次のように設定する。

$$T_{max} = T_{min} + T_{vmax} \cdots$$
 <式 7>

ここで、 T_{vmax} はVUの最大再生時間である。上限値を設定するのは次の理由に基づく。 T_e が大きくなるにしたがって、アフレコ時において図8の(2)から(8)までの期間が長くなる。この間は、再生用データのディスクからの読み出しができないため、再生を継続するためには T_e の増加に応じて、トラックバッファ502のサイズを増やす必要がある。上限値を設定するのは、このとき必要となるトラックバッファ502のサイズを見積り可能にするため

分条件である任意の n における

【0102】

【数1】

なったオーディオの最大ビットレートをそれぞれ R_a 、 R_v 、 R_p としたとき、

$$T_r(i) = T_e(i) \times (R_v + R_a + R_p) / R_s + T_a \cdots$$
 <式 3>

となる。

【0106】右辺第1項はRU# i の読み出し時間を表す。右辺第2項はRU# i 読み出し直後に発生する分断ジャンプによる最大アクセス時間を表す。また、 $T_w(i)$ は、

Bとなる。このような項が必要な理由は、PRUの両端はECCブロック境界と一致しているとは限らないため、PRU記録時には、PRUのサイズより最大2ECCブロック分多く記録することになるためである。ただし、RUの先頭のPRUはECCブロック境界に位置するため、 $(2M-1)$ となっている。

【0111】ここで、 M は $T_e(i)$ に依存するため、 M を $T_e(i)$ で表現することを考える。RU(i)中のVU再生時間の最小値を T_{vmin} とすると、 $M \leq \text{ceiling}(T_e(i) / T_{vmin}) \leq T_e(i) / T_{vmin} + 1$ となる。尚、 $\text{ceiling}(x)$ は x 以上の最小の整数を求める関数である。

【0112】このとき、<式 4>の M に $T_e(i) / T_{vmin} + 1$ を代入しても<式 1>が成立するように $T_e(i)$ を設定すれば、VU再生時間が T_{vmin} 以上であればRUをどのような再生時間のVUで構成しようともリアルタイムアフレコは可能になる。

【0113】<式 1>に<式 3>と<式 4>を代入して、 $T_e(i)$ で解くと、リアルタイムアフレコを保証可能な $T_e(i)$ の条件

限值 T_{min} は、

である。

【0116】また、下限値と上限値の間にVUの最大再生時間分のマージンがあることにより、任意再生時間のVUの組み合わせでRUを構成することが可能である。尚、ここでは最大再生時間をAVストリームのビットレートに応じて設定しているが、可能な最大のビットレートに基づき、AVストリームのビットレートに関わらず一定としても良い。

【0117】尚、本実施例では連続記録単位を再生時間で管理しているが、再生時間にデータのビットレートを乗じた記録領域サイズで管理してもよいのは言うまでもない。

【0118】尚、本実施例では、再生用データ読み出し時に読み出したPRU境界を含むECCブロックをトラックバッファ502からアフレコ用バッファ504に転送してPRU記録時に用いているが、転送せずPRU記録直前にPRU境界を含むECCブロックを読み出しても良い。

$$Tw(i)=2Ta+(M-1) \times Tk+M \times Tr+Te \times Rp/Rs+2 \times (2M-1) \times Ly/Rs \cdots \text{<式 8>}$$

で計算すれば良い。尚、Trは最大回転待ち時間である。

【0120】また、本実施例では、分断ジャンプと過去のRUへのピックアップの移動を非同期に行うことを想定している。この理由は、非同期に行った方が同期して行った場合に比べ、リアルタイムアフレコを行うための条件として厳しい(再生用データの読み出しが途切れる期間が長い)ため、非同期でリアルタイムアフレコが可能であれば同期でも可能であり、実装の自由度を高めることが可能であるためである。

【0121】従って、分断ジャンプと過去のRUへのピックアップの移動を同期して行うことを前提にTeminを設定しても良い。この場合、<式 3>の第2項を取り除いて考えれば良い。

【0122】次に、アフレコ非対応ストリームのディスク上での配置決定方法について説明する。アフレコ対応ストリームと同様、<式 1>を満たせばシームレス再生は保証される。ただし、アフレコデータの記録は行わないため、Twa=0となる。

【0123】Tr(i)は<式 3>と共通であるため、<式 1>に<式 3>およびTwa=0を代入して、Te(i)で解くと、

$$Te(i) \geq Ta/(Rs-Rv-Ra) \cdots \text{<式 9>}$$

が得られる。

【0124】つまり、シームレス再生保証可能なRU再生時間下限値Teminは、

$$Temin \geq Ta/(Rs-Rv-Ra) \cdots \text{<式 10>}$$

となる。

【0125】すなわち、アフレコ対応ストリームと非対応ストリームとの間ではRU再生時間の範囲が異なることになる。なぜなら、非対応ストリームを対応ストリームと同一の範囲にした場合、ディスク上の<式 10>を満たすが<式 6>を満たさない空き連続領域があった場合に、アフレコ非対応ストリームを記録するという選択肢が無くなってしまふからである。

【0126】<バッファサイズ>次に、アフレコ時に必要なトラックバッファ502のサイズについて説明する。尚、ここではショック等の外乱については考慮しない。必要なサイズは、再生データ読み出しからPRU記録に移

$$Lt \geq (Ttmax+2(Twmax+Ta)+Tvmax) \times (Rv+Ra+Rp) \cdots \text{<式 11>}$$

となる。

【0131】ここで、Ttmax=Tvmax \times (Rv+Ra+Rp)/Rs、T

$$Lt \geq (Tvmax \times (Rv+Ra+Rp) + 2((3Ta+(Temax \times Tk)/Tvmin) \times Rs) + 2Temax \times Ly/Tvmin + Temax \times Rp) \times (Rv+Ra+Rp)/Rs \cdots \text{<式 12>}$$

が得られる。

【0119】この場合、PRU記録の際、PRU読み出し時間とピックアップ移動時間とが余分にかかることになる。各PRUについて記録の直前にPRU境界を含むECCブロックの読み出しを行った場合、<式 1>のTw(i)を、

るタイミングの間隔のばらつきが大きいほど増大する。例えば、PRU記録に移るタイミングの間隔が狭い場合が続くようであれば、トラックバッファ502中のデータの消費が続いて発生するため、トラックバッファ502にはより多くの容量が必要である。

【0127】ここでは、任意のRUであるRU#Nのアフレコデータ記録準備完了から、次のRUであるRU#N+1のアフレコデータ記録準備完了するまでの間にRU#Nのアフレコデータ記録処理(図8の(2)~(8))が完了するという場合で考える。この幅は、現在読み出しを行っているRUの読み出しが完了してからアフレコデータ記録を開始可能な値である。

【0128】この条件においては、あるRUのアフレコデータ記録準備完了から次のRUのアフレコデータ記録準備完了までの間のどのタイミングで分断ジャンプやアフレコデータ記録が発生するか不定となる。従って、アフレコデータ記録準備完了直後にアフレコデータ記録1回と分断ジャンプ1回とが連続的に発生しても再生が途切れないように、アフレコデータ記録準備完了時にアフレコデータ記録1回および分断ジャンプ1回を連続的に行っても再生継続が可能なデータが必ずトラックバッファ502になければならない(条件1)。

【0129】また、アフレコデータ記録準備完了直前に、アフレコデータ記録と分断ジャンプとが発生することも考えられる。そうした場合にも上述の条件1を満たすためには、トラックバッファ502はアフレコデータ記録準備完了直前のアフレコデータ記録1回および分断ジャンプ1回の直前にアフレコデータ記録2回および分断ジャンプ2回が連続的に発生したとしても再生継続が可能なデータを保持できなければならない(条件2)。

【0130】つまり、トラックバッファ502はアフレコデータ記録2回および分断ジャンプ2回分の再生時間に対応する容量が必要となる。このほか、次のVUを読み出すまで再生継続するために必要な再生時間Ttmaxおよび、任意のVUの組み合わせであったとしても、必要な再生時間をトラックバッファに確保するためのマージンTvmaxを含めて必要なトラックバッファ502のサイズLtは、

$$wmax=2Ta+Temax \times Tk / Tvmin+(2Temax \times Ly)/(Tvmin \times Rs)+Temax \times Ra/Rs \text{を、<式 11>に代入して、}$$

【0132】次に、アフレコ時に必要となるアフレコ用

バッファ504のサイズについて説明する。アフレコ用バッファ504に関しても、トラックバッファサイズと同様、再生データ読み出しからPRU記録に移るタイミングの間隔のばらつきに影響を受ける。

【0133】ここでは、トラックバッファサイズ試算と同様、任意のRUであるRU#Nのアフレコデータ記録準備完了から、次のRUであるRU#N+1のアフレコデータ記録準備完了するまでの間にRU#Nのアフレコデータ記録処理(図

$$Lp \geq 3 \times (T_{\max} \times R_p + (2 \times (T_{\max} / T_{\min}) - 1) \times L_y) \cdots \text{式13}$$

となる。尚、右辺第2項は、PRU境界を含むECCブロックを保持するためのものである。

【0136】次に、アフレコ非対応ストリームの再生時に必要なトラックバッファ502のサイズについて説明する。尚、ここではショック等の外乱については考慮しない。

【0137】最もトラックバッファが必要になるのは、再生時間 T_{\max} のVUの読み出し中に分断ジャンプが発生

$$L \geq (T_{\max} + T_a + T_{\max}) \times (R_v + R_a) + T_{\max} \times (R_v + R_a) \cdots \text{式14}$$

となる。

【0139】右辺第1項は、新しいVUを読み出すまで再生するために必要なデータ量である。再生時間が T_{\max} 分多いのは、いかなる再生時間のVUがトラックバッファ502中に混在した場合であっても、必要な再生時間をトラックバッファ502中に必ず確保するためである。

$$L \geq (T_{\max} \times (R_v + R_a + 2R_s) + T_a \times R_s) \times (R_v + R_a) / R_s \cdots \text{式15}$$

が得られる。

【0142】＜記録時の処理＞ユーザから録画が指示された場合の処理を、図9のフローチャートに沿って説明する。このとき記録するAVストリームは、ビデオのビットレート $R_v=5\text{Mbps}$ 、オーディオのサンプリング周波数48kHz、ビットレート $R_a=256\text{kbps}$ で、VU再生時間固定のアフレコ対応ストリームであるとする。また、すでにファイルシステムの管理情報はRAM上に読み込まれているものとする。

【0143】まず、ストリームの構成や連続領域の構成を決定する(ステップ701)。1VUを1GOP15フレームで構成するとしたとき、＜式6＞、＜式7＞に $R_s=20\text{Mbps}$ 、 $T_a=1\text{秒}$ 、 $T_k=0.1\text{秒}$ 、 $R_v=5\text{Mbps}$ 、 $R_a=R_p=256\text{kbps}$ 、 $T_{\max}=T_{\min}=\text{約}0.5\text{秒}$ を代入し、 $T_e(i)$ の範囲10.49秒以上10.99秒以下が得られる。

【0144】 $T_{\max}=\text{約}0.5\text{秒}$ でこの条件を満たすのは $T_e(i)=10.5\text{秒}$ のときとなり、21個のVU毎でRUを構成することになる。MPPEG-1 audio layer-IIにおいて、サンプリング周波数が48kHzの場合、AAUあたりの再生時間は0.024秒となるため、1VUには20個か21個のAAUが入る。

【0145】また、オーディオの再生可能な最大ビットレートを256kbpsとしたとき、各AAUの最大サイズは768byteとなる。従って、各PRUの領域サイズは対応するメインオーディオのAAU数 $\times 768\text{byte}$ で確保する。

【0146】PRUの領域サイズを固定値としている理由

8の(2)～(8))が完了するという場合で考える。

【0134】この条件においては、最大RU2個分のPRU記録が連続する可能性がある。さらに、PRU記録の最中にエンコードされるオーディオデータを格納するためのRU1個分のPRU領域が必要となる。従って、RU3個分のPRUブロックを格納する領域があれば良い。

【0135】よって、アフレコ用バッファ504のサイズ Lp は、

する場合である。この間は、トラックバッファに新しいVUが読み出されないため、トラックバッファ中にデータで再生を続ける必要がある。

【0138】つまり、 $T_{\max}+T_a$ 分の再生時間がトラックバッファ502中に存在しなければならない。ここで、 T_{\max} は再生時間 T_{\max} のVUの読み出し時間を表す。このとき必要なトラックバッファ502のサイズ L は、

【0140】右辺第2項は読み出し中のVUを保持するための領域である。この他に、バッファフル時の間欠再生のためのマージンが必要であるが、ここでは省略する。

【0141】ここで、 $T_{\max}=T_{\max} \times (R_v + R_a) / R_s$ を＜式14＞に代入し、

を以下に示す。アフレコ対応ストリームを録画する際、PRUの領域サイズは対応するVUに含まれるAAU数と同じ数のAAUを格納できる必要がある。

【0147】しかし、対応するVUに含まれるAAU数をエンコード前に厳密に知ることは難しい。なぜなら、AAUとGOPの周期は整数倍になっておらず、VU中のオーディオとビデオの開始タイミングは一致するとは限らないためである。そのため、PRUの領域サイズは再生時間 $T_e(i)$ に含まれる最大のAAU数としている。

【0148】このことによって、全体のビットレートが増加するが、オーディオビットレート R_a が256kbpsのとき数秒に最大768byteであり、無視しても良い。また、AVストリーム中に再生に寄与しないデータが含まれることになるが、QuickTime管理情報を用いて再生対象から除外することができるので問題はない。

【0149】次に、21個のVUを連続的に記録可能な空き領域を探す。具体的には $21 \times T_{\max} \times (R_v + R_a + R_p)$ 、つまり57.9Mbit以上の連続的な空き領域をRAM102上のSpace Bitmapを参照して探す。存在しなければ録画を中止し、録画できないことをユーザに知らせる(ステップ702)。

【0150】次に、オーディオエンコーダ117、ビデオエンコーダ118をそれぞれ起動する(ステップ703)。また、記録用バッファに1ECCブロック分(32KB)以上のデータが蓄積されているかどうかをチェックし(ステップ704)、蓄積されている間、ステップ705からステップ708を

繰り返す。

【0151】蓄積されていれば、次に記録するディスク上のECCブロックの空き状況をRAM上のSpace Bitmapを参照して調べる(ステップ705)。空きがなければ、21個のVUを記録可能な連続的な空き領域を探し(ステップ707)、その空き領域の先頭ヘビックアップを移動し(ステップ708)、記録用バッファ111中の1ECCブロック分のデータをディスクに記録する(ステップ706)。

【0152】記録用バッファ111にデータが蓄積されていなければ、記録終了が指示されているかどうかをチェックし(ステップ709)、記録終了でなければステップ704を実行する。

【0153】記録終了が指示されていた場合、以下のステップを実行する。まず、記録用バッファ中の32KBに満たないデータに関して、末尾にダミーデータを付加し32KBにする(ステップ710)。次に、そのデータをディスク上に記録する(ステップ711～ステップ714)。最後に、RAM102上のQuickTime管理情報とファイルシステム管理情報とを光ディスク106に記録する(ステップ715～ステップ716)。

【0154】以上の処理と並行するオーディオエンコーダ117、ビデオエンコーダ118やマルチプレクサ113の動作について説明する。それぞれのエンコーダはマルチプレクサ113にエンコード結果を送り、マルチプレクサはそれらを多重化用バッファ114に格納する。

【0155】1VU分のデータ、つまり1GOPとそれに同期して再生されるAAUが多重化用バッファ114に蓄積されたら、マルチプレクサ113は記録用バッファ111に1VUのデータを送る。その際、マルチプレクサ113はそのVU中のAAUの個数に応じて、最大ビットレートのAAUを格納可能なPRUを多重化する。

【0156】さらに、ホストCPU101に1VU分のデータがエンコードできたことを通知し、ホストCPU101はVUを構成するGOPやAAUの数およびサイズを基にRAM102上のQuickTime管理情報を更新する。

【0157】尚、ここでは、記録したストリームに対して、いずれの機器でもアフレコ可能なように、リファレンス・デバイスの性能(Rs、Ta、Tk)に基づき、RUに含まれる再生時間を決定したが、録画機器の持つ性能に基づきこれを決定しても良い。その場合、アフレコを行う機器がそのAVストリームに対し、アフレコ可能かどうかを判断できるよう、QuickTime管理情報中のX descriptor atomのset performance atomに、性能を格納しておく。

【0158】＜アフレコ時の処理＞ユーザからアフレコが指示された場合の処理を、図10のフローチャートに沿って説明する。ここでは、すでにアフレコの対象となるAVストリームに関するQuickTime管理情報はRAM102に読み込まれているものとする。

【0159】まず、そのQuickTimeムービーが1ファイルのみのアフレコ対応ストリームで構成されているかを調

べ、そうでなければユーザにアフレコできないことを通知する(ステップ801)。これは、独立にディスクに記録されたストリーム同士を非破壊編集したものは、前述したアフレコのための条件を満たす保証がないからである。

【0160】アフレコ開始位置を含むディスク上のVUの先頭から再生用データの読み出しを行う(ステップ802)。このとき、十分な再生時間分のデータを読み出すまでステップ802を繰り返す(ステップ803)。ここで、十分な再生時間分のデータとは、再生用データ読み出しの中断期間が最大の場合でも、再生が途切れなだけのデータ量を意味する。

【0161】また、PRUを読み出した際には、PRUを含むECCブロックをアフレコ用バッファ111に送る。このとき、アフレコ用バッファ111中のPRUを管理するために、アフレコ用バッファ111中の各PRUの再生開始時間(AVストリームの先頭からの相対時間)とアフレコ用バッファ111中でのアドレスの組をテーブルとしてRAM102に保持する。

【0162】次に、ビデオデコーダ116とオーディオデコーダ115、およびオーディオエンコーダ117を起動する(ステップ804)。オーディオエンコーダ117はサンプリングされた音声波形をAAUにエンコードし、AAUの周期でマルチプレクサ113に送る。その際に、各AAUについてAVストリームの先頭からの相対時間を付加する。

【0163】マルチプレクサ113は、AAUに付加された時間に基づき、AAUをアフレコ用バッファ111中のPRUに格納する。RU中の最後のPRUにAAUを最後まで格納し終わったら、ホストCPU101にRUのエンコード終了を通知する。

【0164】次に、ユーザからアフレコ終了を指示されていないがチェックする(ステップ805)。指示されていないならば、PRUのエンコードが終了するまで、ステップ802と同様に再生用データの読み出しを行う(ステップ809)。

【0165】マルチプレクサ部からRUエンコード終了が通知されたら(ステップ806)、RAM102上のテーブルに保持しているそのRUに含まれるPRUの再生開始時間から、QuickTime管理情報を用いそのPRUを記録すべき光ディスク106上のアドレス、つまり元々そのPRUが記録されていたアドレスを求める。そのアドレスにピックアップ107を移動させ(ステップ807)、そのPRUを含むECCブロックを光ディスク107に記録する(ステップ808)。

【0166】アフレコ終了を指示されていれば、現在エンコード中のPRUのエンコード完了を待って(ステップ810)、そのPRUの記録アドレスを求めピックアップを移動し(ステップ811)、PRUを記録する(ステップ812)。最後にQuickTime管理情報をディスクに記録する(ステップ813)。

【0167】尚、本実施例においては、再生用データの読み出しを中断してPRUの記録を開始する際に、再生用

バッファ110の占有量のチェックを行っていないが、アフレコ時のショックへの耐性を高めるためにはチェックを行った方がよいことは言うまでもない。ただし、この場合、PRU記録タイミングが遅れる分、より多くの再生用バッファ110およびアフレコ用バッファ111の容量が必要となる。

【0168】＜再生時の処理＞ユーザから再生が指示された場合の処理を、図11のフローチャートに沿って説明する。ここで、すでに再生の対象となるAVストリームに関するQuickTime管理情報はRAM102に読み込まれているものとする。

【0169】光ディスク107上の再生指示されたVUの先頭から再生用データの読み出しを行う(ステップ901)。このとき、十分な再生時間分のデータを読み出すまでステップ901を繰り返す(ステップ902)。

【0170】ここで、十分な再生時間分のデータとは、再生用データ読み出しの中断期間が最大の場合でも、再生が途切れなだけのデータ量を意味する。具体的には、AVデータの読み出しに伴う分断のジャンプ(最大1秒)を行った場合を想定し、1秒分のデータ量とする。

【0171】次に、ビデオデコーダ116およびオーディオデコーダ115を起動する(ステップ903)。また、ユーザから再生終了を指示されていないかチェックする(ステップ904)。指示されていない場合は、再生用AVデータの読み出しを行う(ステップ905)。再生終了を指示されていれば、終了する。

【0172】＜第2実施例＞次に、本発明における第2の実施例について説明する。上述した第1の実施例との違いは、第1の実施例がPRUのみを書き換えるのに対し、本実施例ではRU全体を書き換える点である。本実施例は第1の実施例と類似するため、相違点に絞って以下説明する。尚、新たに定義していない記号は、第1の実施例における定義を用いる。

【0173】＜AVストリームの形態＞本実施例におけるAVストリームの構成は、上述の第1の実施例と全く同一であるため説明は省略する。

【0174】＜ディスク配置決定方法＞アフレコ非対応ストリームにおけるRU再生時間の決定方法は、上述の第1の実施例と共通である。

【0175】次に、アフレコ対応ストリームにおけるRU再生時間の決定方法について説明する。この決定方法では、第1の実施例と同様、機器間の互換性確保のため、リファレンス・デバイス・モデルおよびリファレンス・アフレコ・アルゴリズムを想定し、それらを用いてアフレコを行った際にシームレス再生が破綻しないようにRU再生時間を決める。

【0176】リファレンス・デバイス・モデルについては、図7とともに上述した第1の実施例のものと同一の構成を取る。ただし、第1の実施例では、トラックバッファ502からアフレコ用バッファ504に転送されるのが、

PRUを含むECCブロックだけであるのに対し、第2の実施例ではRU全体が転送される。また、アフレコデータのディスクへの記録時にはPRUを含むECCブロックだけではなく、RU全体を記録する。

【0177】次に、リファレンス・アフレコ・アルゴリズムについて、図12を用いて説明する。尚、図12中の(1)から(5)までの番号は、以下の説明中の(1)から(5)までの番号に対応する。アルゴリズムの概要は次の通りである。

【0178】(1) 再生用データの読み出しを行う。(2) N番目のRUであるRU#Nに対応するオーディオデータのエンコードが終了すると同時に、RU#Nへのアクセスを行う。(3) アフレコ用バッファ504のデータをRU#Nを記録する。(4) 元の読み出し位置に戻る。(5) 再生用データの読み出しを再開する。以上の動作を繰り返す。

【0179】前記リファレンス・デバイス・モデルにおいて、前記リファレンス・アフレコ・アルゴリズムを用いてアフレコを行った場合、第1の実施例と同様、＜式1＞を満たすことで、アフレコ用バッファ504のオーバーフローおよびトラックバッファ502のアンダーフローがないことが保証できる。

【0180】第2の実施例において、＜式1＞中の $Tr(i)$ は、＜式3＞と共通である。＜式1＞中の $Tw(i)$ は、 $Tw(i)=2Ta+Te(i) \times (Rv+Ra+Rp)/Rs \cdots$ ＜式16＞ここで、右辺第1項は、RU#iへの往復アクセス時間を示す。右辺第3項は、RU#iをディスクに記録するための時間を表す。尚、RUは整数個のECCブロックで構成されているため、＜式4＞の右辺第4項のような項は不要である。

【0181】＜式1＞に＜式3＞と＜式16＞を代入して、 $Te(i)$ で解くと、リアルタイムアフレコを保証可能な $Te(i)$ の条件

$Te(i) \geq 3Ta/(Rs-2(Rv+Ra+Rp)) \cdots$ ＜式17＞が得られる。

【0182】つまり、アフレコ保証可能なRU再生時間下限値 $Temin$ は、

$Temin=3Ta/(Rs-2(Rv+Ra+Rp)) \cdots$ ＜式18＞となる。

【0183】このとき、RU再生時間の上限值 $Temax$ を次のように設定する。

$Temax=Temin+Tvm \cdots$ ＜式19＞

RU再生時間に上限値を設定する理由は、第1の実施例で述べた理由と同様である。また、第1の実施例で説明したのと同様、可能な最大ビットレートに基づき、AVストリームのビットレートに関わらず最大再生時間を一定としても良い。

【0184】尚、本実施例では連続記録単位を再生時間で管理しているが、再生時間にデータのビットレートを乗じた記録領域サイズで管理してもよいのは言うまでもない。

【0185】また、本実施例では、分断ジャンプと過去のRUへのピックアップの移動を非同期に行うことを想定しているが、第1の実施例と同様、同期して行うように想定して、 T_{\min} を設定しても良い。

【0186】＜トラックバッファサイズ＞次に、アフレコ時に必要なトラックバッファ502のサイズについて説明する。尚、ここではショック等の外乱については考慮しない。第1の実施例と同様、必要なサイズは、再生データ読み出しからPRU記録に移るタイミングの間隔のばらつきが大きいくほど増大する。

【0187】ここでは、第1の実施例と同様、任意のRUであるRU#Nのアフレコデータ記録準備完了から、次のRUであるRU#N+1のアフレコデータ記録準備完了するまでの

$$L \geq ((2T_{\max} + T_{\min}) \times (R_v + R_a + R_p) + (T_{\max} + 6T_a) \times R_s) \times (R_v + R_a + R_p) / R_s \cdots \text{式 20}$$

が得られる。

【0190】次に、アフレコ時に必要となるアフレコ用バッファ504のサイズについて説明する。第1の実施例において説明した理由により、アフレコ用バッファ504のサイズは3RU分必要となる。従って、アフレコ用バッファ504のサイズ L_p は、

$$L_p \geq 3 \times T_{\max} \times (R_v + R_a + R_p) \cdots \text{式 21}$$

となる。

【0191】記録時、再生時の処理は、第1の実施例と同様であるため省略する。アフレコ時の処理は、次の点で第1の実施例とは異なる。まず、ステップ803において、第1の実施例ではPRUを読み出した場合のみ、再生用バッファ110からアフレコ用バッファ111へのデータの転送を行なっているが、本実施例では読み出したすべてのデータに関して転送する。

【0192】次に、ステップ808において第1の実施例ではPRUを含むECCブロックのみディスクに記録しているが、本実施例では、アフレコ対象のVUすべてに関して記録する。

【0193】なお、本実施例と第1の実施例との使い分けであるが、実施するシステムにおいて＜式17＞で得られるRU再生下限値と＜式9＞で得られるRU再生下限値を算出し、RU再生下限値が小さくなる方を採用すればよい。なぜなら、RUが短い方が、配置等の制限がつきにくいからである。

【0194】＜第3実施例＞上述した第1及び第2の実施例においては、AVストリームをRU単位にてディスク上で分散して記録することを前提としているが、ピックアップ移動時間の長い機器においては、これらの方法ではアフレコができない場合が考えられる。

【0195】本発明における第3の実施例は、そのような場合にアフレコ対応のAVストリームに限ってディスク上に連続的に配置されるようにすることで、ピックアップ移動時間の最大時間を削減し、ピックアップ移動時間の長い機器でもアフレコを可能にするものである。

間に、RU#Nのアフレコデータ記録処理(図12の(2)～(4))が完了するという場合を考える。尚、ここでのアフレコデータ記録とは、第1の実施例と異なり、RU全体の記録のことを指す。

【0188】この場合に必要なトラックバッファの容量は、第1の実施例において行った説明と同様の理由により、トラックバッファ502はアフレコデータ記録2回および分断ジャンプ2回分の再生時間に対応する容量となる。つまり、必要なトラックバッファ502のサイズ L_t は、＜式11＞で表される。

【0189】ここで、 $T_{t\max} = T_{v\max} \times (R_v + R_a + R_p) / R_s$ 、 $T_{w\max} = 2T_a + T_{\max} \times (R_v + R_a + R_p) / R_s$ を＜式11＞に代入し、

【0196】AVストリームの形態については、上述の第1の実施例と共通である。ただし、アフレコ対応のAVストリームに関しては、1個のAVストリームは1個のRUで構成される。

【0197】＜記録時の処理＞アフレコ対応のAVストリームの記録は、第1の実施例と同様の方法で行う。ただし、第1の実施例においては、連続的に記録可能な空き領域の記録が終了したら次の空き領域に移って記録を継続するのに対し、本実施例では連続的に記録可能な空き領域の1個目の記録が終了した時点で記録を終了する点で異なる。

【0198】＜アフレコ時の処理＞アフレコ処理は、第1の実施例と同様の手順で行う。ただし、RUという単位がないため、アフレコデータ記録を開始するタイミングは次のいずれかの方法で決定する。

【0199】まず、第1の方法について説明する。第1の方法は、第1の実施例と同様、PRUを含むECCブロックのみを記録することを前提とし、＜式6＞で得られる T_{\min} の間隔でアフレコデータ記録を開始するというものである。

【0200】ただし、第1の実施例においては、 T_a はディスクからの読み出し、記録の停止する最大の時間であったが、本実施例では数十トラックから数百トラック程度の SEEK 時間と回転待ち時間とを合わせたものになる。

【0201】次に、第2の方法について説明する。第2の方法は、第2の実施例と同様、アフレコオーディオ以外のメインオーディオ、ビデオデータも書き換えることを前提とし、＜式13＞で得られる T_{\min} の間隔でアフレコデータ記録を開始するというものである。

【0202】ただし、第2の実施例においては、 T_a はディスクからの読み出し、記録の停止する最大の時間であったが、本実施例では数十トラックから数百トラック程度の SEEK 時間と回転待ち時間とを合わせたものになる。

【0203】＜第4実施例＞上述した第1乃至3の実施例においては、アフレコ時にオーディオ入力と並行してディスクへのオーディオデータ記録を行なうことを前提としているが、ピックアップ移動時間の長い機器の場合、そのような制御を行うことは困難である。本発明の第4の実施例は、そのような場合を想定したものである。

【0204】AVストリームの形態については、上述の第1の実施例と共通である。また、RUあたりの再生時間は、アフレコ対応ストリームに関してはアフレコ非対応ストリームと同様、＜式 10＞に基づき決定する。記録時や再生時の処理は、第1の実施例と同様であるので省略する。

【0205】＜アフレコ時の処理＞ユーザからアフレコが指示された場合の処理を、図13のフローチャートに沿って説明する。ここでは、すでにアフレコの対象となるAVストリームに関するQuickTime管理情報はRAM102に読み込まれているものとする。

【0206】まず、アフレコ開始位置を含むディスク上のVUの先頭から再生用データの読み出しを行う(ステップ1301)。このとき、十分な再生時間分のデータを読み出すまでステップ1301を繰り返す(ステップ1302)。ここで、十分な再生時間分のデータとは、再生用データ読み出しの中断期間が最大の場合でも再生が途切れなだけのデータ量を意味する。

【0207】また、PRUを読み出した際には、PRUを含むECCブロック(PRUブロック)をアフレコ用バッファ111に送る。このとき、アフレコ用バッファ111中のPRUを管理するために、アフレコ用バッファ111中の各PRUの再生開始時間(AVストリームの先頭からの相対時間)とアフレコ用バッファ111中でのアドレスの組をテーブルとしてRAM102に保持する。

【0208】次に、ビデオデコーダ116とオーディオデコーダ115、およびオーディオエンコーダ117を起動する(ステップ1303)。オーディオエンコーダ117はサンプリングされた音声波形をAAUにエンコードし、AAUの周期でマルチプレクサ113に送る。

【0209】その際に、各AAUについてAVストリームの先頭からの相対時間を付加する。マルチプレクサ113は、AAUに付加された時間に基づき、AAUをアフレコ用バッファ111中のPRUブロックに格納する。

【0210】次に、ユーザからアフレコ終了を指示されていないかチェックする(ステップ1304)。指示されていないければ、PRUのエンコードが終了するまで、ステップ1301と同様に、再生用データの読み出しを行う(ステップ1305)。アフレコ終了を指示されていれば、アフレコ用バッファ111中に未記録のPRUブロックがある限り(ステップ1306)、以下の処理を行う。

【0211】まず、RAM102上のテーブルに保持しているそのPRUの再生開始時間から、QuickTime管理情報を用い

そのPRUブロックを記録すべき光ディスク106上のアドレス、つまり元々そのPRUブロックが記録されていたアドレスを求める。

【0212】次に、そのアドレスにピックアップ107を移動させ(ステップ1307)、そのPRUを含むECCブロックを光ディスク107に記録する(ステップ1308)。未記録のPRUブロックがなくなったら、最後にQuickTime管理情報をディスクに記録する(ステップ1309)。

【0213】以上のアフレコ処理においては、すべてのPRUをアフレコ後に記録しているが、アフレコ中に記録可能なPRUは記録しておくことで、アフレコ後の待ち時間を削減することが可能である。そのような処理について、図14のフローチャートに沿って説明を行う。ここで、すでにアフレコの対象となるAVストリームに関するQuickTime管理情報はRAM102に読み込まれているものとする。

【0214】まず、アフレコ開始位置を含むディスク上のVUの先頭から再生用データの読み出しを行う(ステップ1401)。このとき、十分な再生時間分のデータを読み出すまでステップ1401を繰り返す(ステップ1402)。ここで、十分な再生時間分のデータとは、再生用データ読み出しの中断期間が最大の場合でも再生が途切れなだけのデータ量を意味する。

【0215】また、PRUを読み出した際には、PRUを含むECCブロック(PRUブロック)をアフレコ用バッファ111に送る。このとき、アフレコ用バッファ111中のPRUを管理するために、アフレコ用バッファ111中の各PRUの再生開始時間(AVストリームの先頭からの相対時間)とアフレコ用バッファ111中でのアドレスの組をテーブルとしてRAM102に保持する。

【0216】次に、ビデオデコーダ116とオーディオデコーダ115、およびオーディオエンコーダ117を起動する(ステップ1403)。オーディオエンコーダ117はサンプリングされた音声波形をAAUにエンコードし、AAUの周期でマルチプレクサ113に送る。その際に、各AAUについてAVストリームの先頭からの相対時間を付加する。

【0217】マルチプレクサ113は、AAUに付加された時間に基づき、AAUをアフレコ用バッファ111中のPRUブロックに格納する。RU中の最後のPRUにAAUを最後まで格納し終わったら、ホストCPU101にRUのエンコード終了を通知する。

【0218】次に、ユーザからアフレコ終了を指示されていないかチェックする(ステップ1404)。指示されていないければ、PRUのエンコードが終了するまで、ステップ1301と同様に再生用データの読み出しを行う(ステップ1409)。

【0219】マルチプレクサ部からPRUエンコード終了が通知されたら(ステップ1405)、そのPRUを記録可能かどうかを判断する(ステップ1406)。具体的には、現在のピックアップの位置とそのPRUの位置関係からそのPRU記

録に要する時間を試算し、現在の再生用バッファ残量と比較を行い、PRU記録によって再生用バッファが空あるいは空に近くならないようなら、記録可能と判断する。

【0220】記録可能と判断された場合、RAM102上のテーブルに保持しているそのRUに含まれるPRUの再生開始時間から、QuickTime管理情報を用いそのPRUを記録すべき光ディスク106上のアドレス、つまり元々そのPRUが記録されていたアドレスを求める。そのアドレスにピックアップ107を移動させ(ステップ1407)、そのPRUを含むECCブロックを光ディスク107に記録する(ステップ1408)。

【0221】アフレコ終了が指示されていれば、アフレコ用バッファ111中に未記録のPRUブロックがある限り(ステップ1410)、以下の処理を行う。まず、RAM102上のテーブルに保持しているそのPRUの再生開始時間から、QuickTime管理情報を用いてそのPRUブロックを記録すべき光ディスク106上のアドレス、つまり元々そのPRUブロックが記録されていたアドレスを求める。

【0222】次に、そのアドレスにピックアップ107を移動させ(ステップ1411)、そのPRUを含むECCブロックを光ディスク107に記録する(ステップ1412)。未記録のPRUブロックがなくなったら、最後にQuickTime管理情報をディスクに記録する(ステップ1413)。

【0223】＜第5実施例＞上述した第1乃至3の実施例は、アフレコ時に入力されたオーディオデータを、対応するPRUに直接記録しているが、ピックアップ移動時間の長い機器では、そのような制御を行うことは困難な場合がある。本発明における第5の実施例は、そのような場合を想定して、アフレコ時にはディスク上の一時的な領域に記録し、アフレコ後に本来のPRUに移動するというものである。

【0224】AVストリームの形態については、上述の第1の実施例と共通である。また、RUあたりの再生時間はアフレコ対応ストリームに関しては、アフレコ非対応ストリームと同様、＜式10＞に基づき決定する。記録時や再生時の処理は、第1の実施例と同様であるので省略する。

【0225】＜アフレコ時の処理＞ユーザからアフレコが指示された場合の処理を、図15のフローチャートに沿って説明する。ここでは、すでにアフレコの対象となるAVストリームに関するQuickTime管理情報はRAM102に読み込まれているものとする。

【0226】まず、アフレコ開始位置を含むディスク上のVUの先頭から再生用データの読み出しを行う(ステップ1501)。このとき、十分な再生時間分のデータを読み出すまでステップ1501を繰り返す(ステップ1502)。ここで、十分な再生時間分のデータとは、再生用データ読み出しの中断期間が最大の場合でも再生が途切れなだけのデータ量を意味する。

【0227】また、PRUを読み出した際には、PRUを含むECCブロック(PRUブロック)をアフレコ用バッファ111に

送る。このとき、アフレコ用バッファ111中のPRUを管理するために、アフレコ用バッファ111中の各PRUの再生開始時間(AVストリームの先頭からの相対時間)とアフレコ用バッファ111中でのアドレスの組をテーブルとしてRAM102に保持する。

【0228】次に、ビデオデコーダ116とオーディオデコーダ115、およびオーディオエンコーダ117を起動する(ステップ1503)。オーディオエンコーダ117はサンプリングされた音声波形をAAUにエンコードし、AAUの周期でマルチプレクサ113に送る。その際に、各AAUについてAVストリームの先頭からの相対時間を付加する。

【0229】マルチプレクサ113は、AAUに付加された時間に基づき、AAUをアフレコ用バッファ111中のPRUブロックに格納する。マルチプレクサ113は、AAUに付加された時間に基づき、AAUをアフレコ用バッファ111中のPRUに格納する。

【0230】次に、ユーザからアフレコ終了を指示されていないかチェックする(ステップ1504)。指示されていないならば、ピックアップ107を次のVUに移動し(ステップ1505)、ステップ802と同様に、再生用データの読み出しを行う(ステップ1506)。

【0231】アフレコ用バッファ111中に現在あるPRUブロックがアフレコ開始からN個目以降であった場合(ステップ1507)、ピックアップ107を直前に読み出したVUの先頭に移動し(ステップ1508)、アフレコ用バッファ111中の最も古いPRUブロックを一時的に記録する(ステップ1509)。

【0232】その際に、どのPRUをどこに一時的に記録したかをRAM102に記録する。尚、上述のNは再生用データを読み出してからその再生用データに対するアフレコデータが記録可能になる最大の時間に基づいて決定する。

【0233】アフレコ終了が指示されていれば、一時的に記録したPRUが残っている限り(ステップ1510)、次の処理を行う。まず、一時的に記録したPRUの位置にピックアップ107を移動し(ステップ1511)、PRUブロックを読み出し(ステップ1512)、そのPRUを本来記録すべきPRUの位置にピックアップ107を移動し(ステップ1513)、PRUブロックを記録する(ステップ1514)。

【0234】尚、ここではPRU毎に読み書きを行っているが、複数のPRUに関して連続的に読み出しや記録を行っても良いことは言うまでもない。

【0235】一時的に記録したPRUの本来の場所への移動が終わったら、次に、アフレコ用バッファ111にPRUブロックが残っている限り(ステップ1515)、次の処理を行う。PRUブロックに付加されているPRUの再生開始時間から、QuickTime管理情報を用い元々そのPRUブロックの記録されていたアドレスを求め、そのアドレスにピックアップ107を移動させ(ステップ1516)、そのPRUブロックを記録する(ステップ1517)。未記録のPRUブロックがなく

なったら、最後にQuickTime管理情報をディスクに記録する(ステップ1518)。

【0236】本実施例では、PRUブロックを一時的に記録する領域として、ディスクに空き領域が全くない場合を考慮して、同一ストリーム内の別の領域を用いているが、現在の読み出し位置の近傍であれば、同一ストリーム外でも構わないことは明らかである。

【0237】また、本実施例においては、PRUブロックを一時的に記録する媒体として、アフレコ用バッファのような半導体メモリや光ディスクを用いているが、ハードディスク等それ以外の記録媒体であっても構わないことは言うまでもない。

【0238】＜第6実施例＞次に、本発明における第6の実施例について、図16とともに説明する。

【0239】＜AVストリームの形態＞まず、本実施例におけるAVストリームの構成について、図16を用いて説明する。AVストリームには、アフレコ対応ストリームとアフレコ非対応ストリームとがある。アフレコ非対応ストリームは、1秒程度のビデオとオーディオとを多重化したユニットで構成される。ここでは、そのユニットのことをVideo Unit(VU)と呼ぶ。

【0240】ここで、VUの構成については、上述した第1の実施例におけるアフレコ非対応ストリームにおけるVUと共通であるため説明は省略する。

【0241】一方、アフレコ対応ストリームは、図16に示すように、アフレコ非対応ストリームに対し整数個のVU毎にアフレコデータを格納するための領域であるPost Recording Unit(PRU)を多重化したものである。PRUには、次のPRUまでの間のVUと同じ時間に再生されるアフレコ用オーディオデータを格納する。

【0242】PRUが対応するVUより前に位置している、つまりファイル中での相対ブロック番号が小さいのは、VU中のAVデータとPRU中のアフレコ用オーディオデータとを同期再生する際の遅延用バッファが少なくすむからである。

【0243】ここでは、同期再生されるPRUと整数個のVUとをあわせてEditable Unit(EU)と呼ぶ。PRUを多重化する間隔はアフレコの実現性と関連しており、多重化間隔の決定方法については後述する。

【0244】EU内(EU先頭も含む。ただし、AVストリームの先頭は除く)の分断は最大1回までとし、PRUはディスク上で連続的に記録するものとする。これによって、後述するようにEU再生時間の下限を下げるができる。

【0245】PRUの領域サイズの決定方法については、第1の実施例と共通であるため説明を省略する。

【0246】＜ディスク配置決定方法＞次に、アフレコ

$$Tr(i) = Te(i) \times (Rv + Ra) / Rs + Te(i) \times Rp / Rs + Ta \cdots \text{＜式22＞}$$

となる。

【0255】右辺第1項、第2項はそれぞれEU中のVU読み

対応ストリームにおけるPRUを多重化する間隔の決定方法について説明する。この決定方法では、第1の実施例と同様、基準となるデバイス(リファレンス・デバイス・モデル)と、基準となるアフレコアルゴリズム(リファレンス・アフレコ・アルゴリズム)とを想定し、次にそれらを用いてアフレコを行った際にシームレス再生が破綻しないように多重化間隔を決める。

【0247】ここで、リファレンス・デバイス・モデルについては、第1の実施例と共通であるため、説明を省略する。

【0248】次に、リファレンス・アフレコ・アルゴリズムについて、図17を用いて説明する。尚、図17中の(1)から(11)までの番号は、以下の説明中の(1)から(11)までの番号に対応する。アルゴリズムの概要は次の通りである。

【0249】(1) 再生用データの読み出しを行う。(2) N番目のPRUであるPRU(N)に対応するオーディオデータのエンコードが終了すると同時に、PRU(N)へのアクセスを行う。(3) PRU(N)をディスクに記録する。(4) 元の読み出し位置に戻る。

【0250】(5) 再生用データの読み出しを行う。(6) N+1番目のPRUであるPRU(N+1)に対応するオーディオデータのエンコードが終了すると同時に、PRU(N+1)へのアクセスを行う。(7) PRU(N+1)をディスクに記録する。

(8) 元の読み出し位置に戻る。(9) 再生用データの読み出しを行う。(10) 再生用データに分断点があった場合、次の連続領域の先頭にシークする。(11) 再生用データの読み出しを再開する。以上の動作を繰り返す。

【0251】前記リファレンス・デバイス・モデルにおいて、前記リファレンス・アフレコ・アルゴリズムを用いてアフレコを行った場合、次のような条件を満たせば、アフレコ用バッファ504のオーバーフローおよびトラックバッファ502のアンダーフローがないことが保証できる。

【0252】その条件とは、AVストリーム中の任意のEUであるEU#iについて最大再生時間を $Te(i)$ 、分断ジャンプを含めた最大読み出し時間を $Tr(i)$ 、EU#i中のPRUであるPRU#iの最大記録時間を $Tw(i)$ としたとき、第1の実施例と同様、＜式1＞が成立することである。

【0253】このとき、PRUエンコード完了に同期してアフレコデータのディスクへの記録を行っているため、アフレコ用バッファ504中のデータが累積していくことはなく、アフレコ用バッファ504のオーバーフローもない。

【0254】＜式1＞中の $Tr(i)$ は、

出し時間およびPRU読み出し時間を表す。右辺第3項は読み出しに伴う分断ジャンプによるアクセス時間を表す。

EU中には分断が最大1回であるため、1回分のアクセス時間となっている。

【0256】また、 $T_w(i)$ は、

$$T_w(i) = 2Ta + T_e(i) \times R_p / R_s + T_y \cdots \text{＜式23＞}$$
 となる。

【0257】ここで、右辺第1項はPRUへの往復アクセス時間を示す。PRUへの往復のアクセス時間に最大アクセス時間 T_a を用いているのは、第1の実施例と同様の理由に基づく。

【0258】右辺第2項は、PRUをディスクに記録するための時間を表す。右辺第3項である T_y はPRU両端が含まれるECCブロック中のアフレコデータ以外の最大記録時間を表しており、 $T_y = 2 \times 32\text{KB} / R_s$ となる。このような項が必要な理由は、PRUの両端はECCブロック境界と一致しているとは限らないため、PRU記録時には、PRUのサイズより最大2ECCブロック分多く記録することになるためである。

【0259】尚、前述のようにPRUをディスク上で連続的に記録するようにしているため、PRU記録中のアクセスは発生しない。このことにより、PRU記録に伴う時間を短くすることができ、結果としてEU再生時間の下限値を低く抑えることが可能になる。

【0260】＜式1＞に＜式22＞と＜式23＞を代入して $T_e(i)$ で解くと、アフレコを保証可能な $T_e(i)$ の条件

$$T_e(i) \geq ((3Ta + T_y) \times R_s) / (R_s - R_v - R_a - 2Rp) \cdots \text{＜式24＞}$$
 が得られる。

【0261】つまり、アフレコ保証可能なEU再生時間下限値 T_{emin} は、

$$T_{emin} = ((3Ta + T_y) \times R_s) / (R_s - R_v - R_a - 2Rp) \cdots \text{＜式25＞}$$
 となる。

【0262】このとき、EU再生時間の上限値 T_{emax} を次のように設定する。

$$T_{emax} = (3Ta \times R_s) / (R_s - R_v - R_a - 2Rp) + T_{vmax} \cdots \text{＜式26＞}$$

$$Lt \geq (3T_{emax} \times R_p + T_{vmax} \times (R_v + R_a) + (6Ta + 2T_y + T_{vmax}) \times R_s) \times (R_v + R_a + R_p) / R_s \cdots \text{＜式27＞}$$

となる。

【0268】なお、本実施例では、第1の実施例にて説明した理由により、分断ジャンプと過去のRUへのピックアップの移動を非同期に行うことを想定しているが、分断ジャンプと過去のRUへのピックアップの移動を同期して行うことを前提に T_{emin} を設定しても良い。この場合、＜式22＞の右辺第3項を取り除いて考えればよい。トラックバッファサイズに関しては、＜式11＞中の T_a の項を取り除けばよい。

【0269】また、本実施例においては、リファレンス・アフレコ・アルゴリズムとしてPRUの含まれるECCブロックのみ記録しているが、第2の実施例のように、AVストリーム全体を再記録するようリファレンス・アフレコ・アルゴリズムを用いてもよい。その場合、＜式23＞の右辺第2項が $T_e(i) \times (R_v + R_a + R_p) / R_s$ となる。トラック

ここで、 T_{vmax} はVUの最大再生時間である。上限値を設定するのは、第1の実施例で説明した理由に基づく。また、EU再生時間が上記の制限を満たせば、ストリーム中のVU再生時間は固定でも可変でも構わない。

【0263】また、本実施例においては、EU(EU先頭も含む)中での分断回数を最大1回にしているが、任意の回数 N にしても構わない。このことによって、連続領域長を相対的に短くできるため、配置の自由度が高まる、という利点がある。ただし、＜式3＞右辺第3項の T_a に N を乗ずるように変更する必要がある。

【0264】さらに、本実施例では、EU中(EU先頭も含む)での分断回数を最大1回にしているが、AVストリームを構成する各連続領域に必ず1回以上EUの先頭、すなわちPRUが含まれるというように制限しても良い。あるいは、各連続領域には必ず完全なEUが含まれるというように制限しても良い。

【0265】また、VU再生時間がストリーム内で固定の場合、各連続領域に含まれるVUの最小個数で制限しても良い。あるいは、ストリーム内でEU再生時間が固定値 T_e の場合、連続領域長を $M \times T_e \times (R_v + 2Ra)$ 以上に制限しても良い。なお、 M は1以上の整数である。

【0266】そしてまた、本実施例ではEUの再生時間を設定しているが、分断の位置をEUの先頭に限定した場合、連続領域の再生時間を設定していることになり、第1の実施例と比べてAVストリーム構成の異なる1バリエーションと考えることもできる。

【0267】＜バッファサイズ＞次に、アフレコ時に必要なトラックバッファ502のサイズについて説明する。考え方は第1の実施例と同様であるが、AVストリーム構成の違いから＜式11＞中の T_{tmax} および T_{wmax} がそれぞれ

$$T_{tmax} = T_{emax} \times R_p / R_s + T_{vmax} \times (R_v + R_a) / R_s, T_{wmax} = 2Ta + T_y + T_{emax} \times R_p / R_s$$
 となり、必要なトラックバッファサイズ L_t は、

バッファサイズについては、＜式11＞中の T_{wmax} を $2Ta + T_y + T_{emax} \times (R_v + R_a + R_p) / R_s$ とすればよい。

【0270】＜記録時の処理＞次に、ユーザから録画が指示された場合の処理を説明する。処理のフローは第1の実施例と同じであるため、図9を用いて説明する。このとき記録するAVストリームは、ビデオのビットレート $R_v = 5\text{Mbps}$ 、オーディオのビットレート $R_a = 256\text{kbps}$ で、VU再生時間固定のアフレコ対応ストリームであるとする。また、すでにファイルシステムの管理情報はRAM上に読み込まれているものとする。

【0271】まず、ストリームの構成や連続領域の構成を決定する(ステップ701)。1VUを1GOP15フレームで構成するとしたとき、＜式25＞、＜式26＞に $R_s = 20\text{Mbps}$ 、 $T_a = 1\text{秒}$ 、 $R_v = 5\text{Mbps}$ 、 $R_a = 256\text{kbps}$ 、 $T_{vmax} = \text{約}0.5\text{秒}$ を代入し、 $T_e(i)$ の範囲4.22秒以上4.72秒以下が得られる。 T_{vm}

ax=約0.5秒でこの条件を満たすのは $T_e(i)=4.5$ 秒のときとなり、9個のVU毎にPRUが挿入されることになる。

【0272】MPEG-1 audio layer-IIにおいて、ビットレート256kbpsのとき、AAUの再生時間 T_{af} は0.024秒、サイズは768byteとなり、このときのPRUの領域サイズは、144384byteとなる。また、連続領域には9個のVUが含まれるようにする。

【0273】9個のVUと1個のPRUとを連続的に記録可能な空き領域を探す。具体的には $9 \times T_{vmax} \times (R_a + R_v) + 9 \times T_{vmax} \times R_a$ 、つまり24.8Mbit以上の連続的な空き領域をRAM102上のSpace Bitmapを参照して探す。存在しなければ録画を中止し、録画できないことをユーザに知らせる(ステップ702)。

【0274】次に、オーディオエンコーダ117、ビデオエンコーダ118をそれぞれ起動する(ステップ703)。また、記録用バッファに1ECCブロック分(32KB)以上のデータが蓄積されているかどうかをチェックし(ステップ704)、蓄積されている間ステップ705からステップ708を繰り返す。

【0275】蓄積されていれば、次に記録するディスク上のECCブロックの空き状況をRAM上のSpace Bitmapを参照して調べる(ステップ705)。空きがなければ、9個のVUとPRUを記録可能な連続的な空き領域を探して(ステップ707)、その空き領域の先頭ヘビクアップを移動し(ステップ708)、記録用バッファ111中の1ECCブロック分のデータをディスクに記録する(ステップ706)。

【0276】一方、記録用バッファ111にデータが蓄積されていなければ、記録終了が指示されているかどうかをチェックし(ステップ709)、記録終了でなければステップ704を実行する。

【0277】記録終了が指示されていた場合、以下のステップを実行する。まず、記録用バッファ中の32KBに満たないデータに関して、末尾にダミーデータを付加し32KBにする(ステップ710)。次に、そのデータをディスク上に記録する(ステップ711～ステップ714)。さらに、RAM102上のQuickTime管理情報とファイルシステム管理情報とを光ディスク106に記録する(ステップ715～ステップ716)。

【0278】以上の処理と並行するオーディオエンコーダ117、ビデオエンコーダ118やマルチプレクサ113の動作について説明する。それぞれのエンコーダはマルチプレクサ113にエンコード結果を送り、マルチプレクサはそれらを多重化用バッファ114に格納する。

【0279】1VU分のデータ、つまり1GOPとそれに同期して再生されるAAUが多重化用バッファ114に蓄積されたら、マルチプレクサ113は記録用バッファ111に1VUのデータを送る。このとき、そのVUが $9 \times i$ 番目(i は0以上の整数)のVUであったら、上述のサイズを持ったPRUを先に記録用バッファ111に送る。

【0280】さらに、ホストCPU101に1VU分のデータが

エンコードできたことを通知し、ホストCPU101はVUを構成するGOPやAAUの数およびサイズを基に、RAM102上のQuickTime管理情報を更新する。

【0281】＜アフレコ時の処理＞次に、ユーザからアフレコが指示された場合の処理を説明する。処理のフローは第1の実施例と同じであるため、図10を用いて説明する。ここでは、すでにアフレコの対象となるAVストリームに関するQuickTime管理情報はRAM102に読み込まれているものとする。

【0282】まず、そのQuickTimeムービーが1ファイルのみのアフレコ対応ストリームで構成されているかを調べ、そうでなければユーザにアフレコできないことを通知する(ステップ801)。これは、独立にディスクに記録されたストリーム同士を非破壊編集したものは前述したアフレコのための条件を満たす保証がないからである。

【0283】アフレコ開始位置を含むディスク上のPRUの先頭から再生用データの読み出しを行う(ステップ802)。このとき、十分な再生時間分のデータを読み出すまでステップ802を繰り返す(ステップ803)。

【0284】ここで、十分な再生時間分のデータとは、再生用データ読み出しの中断期間が最大の場合でも再生が途切れないだけのデータ量を意味する。具体的には、PRUの記録(最悪約3秒)とAVデータの読み出しに伴う分断のジャンプ(最悪1秒)を連続的に行った場合を想定し、4秒分のデータ量とする。

【0285】また、PRUを読み出した際には、PRUを含むECCブロックをアフレコ用バッファ111に送る。このとき、アフレコ用バッファ111中のPRUを管理するために、アフレコ用バッファ111中の各PRUの再生開始時間(AVストリームの先頭からの相対時間)とアフレコ用バッファ111中でのアドレスの組をテーブルとしてRAM102に保持する。

【0286】次に、ビデオデコーダ116とオーディオデコーダ115、およびオーディオエンコーダ117を起動する(ステップ804)。オーディオエンコーダ117はサンプリングされた音声波形をAAUにエンコードし、AAUの周期でマルチプレクサ113に送る。その際に、各AAUについてAVストリームの先頭からの相対時間を付加する。

【0287】マルチプレクサ113は、AAUに付加された時間に基づき、AAUをアフレコ用バッファ111中のPRUに格納する。PRUにAAUを最後まで格納し終わったら、ホストCPU101にPRUのエンコード終了を通知する。次に、ユーザからアフレコ終了を指示されていないかチェックする(ステップ805)。指示されていないならば、PRUのエンコードが終了するまで、ステップ802と同様に再生用データの読み出しを行う(ステップ809)。

【0288】アフレコ用バッファ111中のあるPRUのエンコードが終了したら(ステップ806)、RAM102上のテーブルに保持しているそのPRUの再生開始時間から、QuickTime管理情報を用いてそのPRUを記録すべき光ディスク106

上のアドレス、つまり元々そのPRUが記録されていたアドレスを求める。そのアドレスにピックアップ107を移動させ(ステップ807)、そのPRUを含むECCブロックを光ディスク107に記録する(ステップ808)。

【0289】一方 アフレコ終了を指示されていれば、現在エンコード中のPRUのエンコード完了を待って(ステップ810)、そのPRUの記録アドレスを求めてピックアップを移動し(ステップ811)、PRUを記録する(ステップ812)。最後に、QuickTime管理情報をディスクに記録する(ステップ813)。

【0290】＜再生時の処理＞次に、ユーザから再生が指示された場合の処理を説明する。処理のフローは第1の実施例と同じであるため、図11を用いて説明する。ここで、すでに再生の対象となるAVストリームに関するQuickTime管理情報は、RAM102に読み込まれているものとする。

【0291】アフレコ開始位置を含む光ディスク107上のPRUの先頭から再生用データの読み出しを行う(ステップ901)。このとき、十分な再生時間分のデータを読み出すまでステップ901を繰り返す(ステップ902)。

【0292】ここで、十分な再生時間分のデータとは、再生用データ読み出しの中断期間が最大の場合でも再生が途切れなだけのデータ量を意味する。具体的には、AVデータの読み出しに伴う分断のジャンプ(最大1秒)を行った場合を想定し、1秒分のデータ量とする。

【0293】次に、ビデオデコード116およびオーディオデコード115を起動する(ステップ903)。また、ユーザから再生終了を指示されていないかチェックする(ステップ904)。指示されていない場合は、再生用AVデータの読み出しを行う(ステップ905)。再生終了を指示されていれば、終了する。

【0294】＜第7実施例＞次に、本発明における第7の実施例について、図18乃至図20とともに説明する。第1乃至6の実施例と第7の実施例との違いは、アフレコデータを格納する領域をAVストリームに多重化しない点にある。また、サブオーディオデータの記録を、ビデオデータおよびメインオーディオデータを再生しながら行うだけでなく、独立に行うことも想定している点が異なる。具体的には、別途に記録したBGMデータを非破壊編集によって、ビデオデータおよびメインオーディオデータと同期再生させることを考慮している点が異なる。

【0295】＜AVストリームの形態＞本実施例におけるAVストリームの構成を説明する。AVストリームには、オーディオとビデオとを多重化したストリーム(AV多重化ストリーム)と、オーディオデータのみで構成されるストリーム(オーディオストリーム)とが存在する。AV多重化ストリームの構成は、上述の第1の実施例におけるアフレコ非対応ストリームと同一であるため、ここでは説明を省略する。

【0296】オーディオストリームにはアフレコオーディオデータが格納され、整数個のAAUで構成される。AV多重化ストリームとオーディオストリームとはそれぞれ別ファイルに格納するが、同一ファイル中に格納しても構わない。なお、分断の位置をVUの先頭に限定した場合、ここでの連続領域は前述のRUと等価になる。

【0297】＜ディスク配置決定方法＞次に、上述のAVストリームをディスク上の複数の連続領域に分散して配置する際の、各連続領域の構成の決定方法について説明する。第1、第2、第5の実施例ではアフレコの際に再生が途切れないように連続領域の構成を決めるのに対し、この決定方法では、基準となるデバイス(リファレンス・デバイス・モデル)および基準となる再生アルゴリズム(リファレンス・プレイバック・アルゴリズム)を想定し、AV多重化ストリームに同期してオーディオストリームを再生したとしても、ビデオやオーディオが途切れることがないように連続領域の構成を決める。

【0298】その理由を以下に説明する。第1、第2、第5の実施例のように、アフレコオーディオデータを記録するための領域を録画時にAVストリームに多重化した場合、ビデオデータとメインオーディオデータおよびアフレコオーディオデータを同期再生する際には、AV多重化ストリームの先頭から順に読み出していけばよい。

【0299】それに対し、本実施例のように、AV多重化ストリーム中にアフレコオーディオデータが多重化されていない場合、ビデオデータとメインオーディオデータおよびアフレコオーディオデータを同期再生する際にも、アフレコ時と同様、AV多重化ストリームとアフレコオーディオデータ間を往復する必要がある。

【0300】さらに、本実施例では、別々に記録したAV多重化ストリームとオーディオストリームを非破壊編集によって同期再生させることも想定しており、記録の自由度が高い。そのため、再生時の方がアフレコ時よりビデオやオーディオを途切れさせず再生するための条件として厳しい。したがって、アフレコ時でなく再生時を基準として連続領域を決めなければならない。

【0301】リファレンス・デバイス・モデルについては、図7とともに上述した第1の実施例のものと同一であるため、リファレンス・プレイバック・アルゴリズムについてののみ、図18を用いて説明する。尚、図18中の(1)から(6)までの番号は、以下の説明中の(1)から(6)までの番号に対応する。

【0302】アルゴリズムの概要は次の通りである。
(1) 再生用データの読み出しをAV多重化ストリーム1001から行う。(2) N個のVUに相当する再生時間を持つオーディオストリーム1002のデータのデコードが終了すると同時に、オーディオストリーム1002へのアクセスを行う。アクセス位置は前回オーディオストリーム1002の読み出しを終了した箇所である。

【0303】(3) N個のVUに相当する再生時間を持つAAU

を読み出す。(4) AV多重化ストリーム1001中の元の読み出し位置に戻る。(5) 再生用データの読み出しを行う。(6) N個のVUに相当する再生時間を持つオーディオストリーム1002のデータのデコードが終了すると同時に、オーディオストリーム1002へのアクセスを行う。以上の動作を繰り返す。

【0304】前記リファレンス・デバイス・モデルにおいて、前記リファレンス・アフレコ・アルゴリズムを用いてアフレコを行った場合、次のような条件を満たせば、再生用バッファのアンダーフローがないことが保証できる。

【0305】その条件とは、N個のVUを表示する間に、常にN個のVUを読み出し、なおかつ、N個のVUに相当するオーディオデータを読み出すことができることである。

【0306】つまり、 T_{av} をVUあたりの再生時間、 Tr_N をN個のVUを読み出すのに必要な時間、 Tra はN個のVUに相当する再生時間を持つオーディオデータの読み出しに必要な時間としたとき、次の式を満たす必要がある。

$$N \times T_{av} \geq Tr_N + Tra \quad \cdots \text{式29}$$

まず、 Tr_N について説明する。

【0307】読み出しにおいては、分断のジャンプ時間も考慮する必要がある。ここでは、連続領域にはN個以上のVUが含まれるようにする。これにより、N個のVUに相当する再生時間の読み出しに要する分断ジャンプは最大1回となる。ここで、 Tr_N はVU中のメインオーディオとビデオの最大ビットレートをそれぞれ R_a 、 R_v としたとき、 $Tr_N = Ta + N \times T_{av} \times (Ra + Rv)$ となる。

【0308】次に、 Tra について説明する。オーディオストリーム中のデータ読み出しにかかる時間は、(オーディオストリームへの往復アクセス時間)+(オーディオデータ読み出し時間)+(オーディオストリーム内での分断ジャンプのためのアクセス時間 $\times M$)となる。ここでMは、オーディオストリーム読み出し中の分断ジャンプ回数を示す。

【0309】オーディオストリーム中のオーディオデータのビットレートを R_p とした場合、 $Tra = 2Ta + N \times T_{av} \times R_p / (Rs + Ta \times M)$ となる。ここで、オーディオストリームを構成する各連続領域には、N個のVUに相当する再生時間以上のAAUが含まれるようにすれば、 $M=1$ となり、 $Tra = 3Ta + N \times T_{av} \times R_p / Rs$ となる。

【0310】 Tra と Tr_N を<式29>に代入し、Nでまとめると次の式が得られる。

$$N \geq (4Ta \times Rs) / (T_{av} \times (Rs - Rv - Ra - R_p)) \quad \cdots \text{式30}$$

つまり、AV多重化ストリームを構成する各連続領域は、それぞれ<式30>を満たすN以上のVUで構成する必要がある。この式を変形すると、 $N \times T_{av} \geq (4Ta \times Rs) / (Rs - Rv - Ra - R_p)$ となる。連続領域の再生時間 Te は、 $Te = N \times T_{av}$ であるため、 $Te \geq (4Ta \times Rs) / (Rs - Rv - Ra - R_p)$ となる。

【0311】一方、オーディオストリームを構成する各連続領域は、同期再生するAV多重化ストリームにおける

N個以上のVUに相当する再生時間を持てば良い。ただし、オーディオストリームは任意のAV多重化ストリームと組み合わせて同期再生される可能性があるため、オーディオストリームを構成する各連続領域は余裕を持った再生時間を持つ必要がある。

【0312】その再生時間 Tc は、具体的にはオーディオおよびビデオの最大ビットレートを R_{amax} 、 R_{vmax} としたとき、以下の条件を満たす必要がある。

$$Tc \geq (4Ta \times Rs) / (Rs - R_{vmax} - 2R_{amax}) \quad \cdots \text{式31}$$

<バッファサイズ>次に、AV多重化ストリームとオーディオストリームの同期再生時に必要なトラックバッファ502のサイズについて説明する。尚、アフレコ時に必要なサイズについては、第1の実施例に基づいて設定する。

【0313】ビデオデータおよびメインオーディオデータに必要なサイズ $Lt1$ とサブオーディオデータに必要なサイズ $Lt2$ に分けて説明する。

【0314】まず、メインオーディオデータとサブオーディオデータに必要なサイズに関して説明する。メインオーディオデータとサブオーディオデータに関して、もっともデータの読み出しが途切れるのは、オーディオストリーム読み出し中に分断ジャンプが1回発生し、なおかつ、AV多重化ストリーム読み出しに戻った直後に分断ジャンプが発生する場合と考えられる。

【0315】したがって、最低限その間再生が継続できるだけのトラックバッファ502のサイズを用意する必要がある。そのサイズ $Lt1$ は、

$$Lt1 \geq (Tra + Ta + T_{av}) \times (Rv + Ra) \quad \cdots \text{式32}$$

となる。

【0316】次に、サブオーディオデータに必要なサイズに関して説明する。サブオーディオデータに関して、もっともデータの読み出しが途切れるのは、AV多重化ストリーム読み出し中に分断ジャンプが1回発生し、なおかつ、オーディオストリーム読み出しに戻った直後に分断ジャンプが発生する場合と考えられる。

【0317】したがって、最低限その間再生が継続できるだけのトラックバッファ502のサイズを用意する必要がある。そのサイズ $Lt2$ は、

$$Lt2 \geq (Tr_N + Ta + T_{av} + 2 \times Ta) \times (Rv + Ra) \quad \cdots \text{式33}$$

となる。

【0318】<記録時の処理>次に、ユーザから録画が指示された場合の処理を説明する。処理の流れは、第1の実施例において説明した図9のものと同一であるため、相違点のみを説明する。このとき記録するAV多重化ストリームは、ビデオのビットレート $R_v = 5\text{Mbps}$ 、オーディオのビットレート $R_a = 256\text{kbps}$ であるとする。また、すでにファイルシステムの管理情報はRAM上に読み込まれているものとする。

【0319】まず、連続領域の構成を設定する(ステップ701)。1VUを2GOP30フレームで構成したとすると、<

式30>に $R_s=20\text{Mbps}$ 、 $T_a=1\text{秒}$ 、 $R_v=5\text{Mbps}$ 、 $R_a=256\text{kbps}$ 、 T_{av} =約1秒を代入し、 $N \geq 5.43$ となる。そこで、 $N=6$ とする。つまり、各連続領域には6個のVUが含まれるようにする。

【0320】また、ステップ702、ステップ707では6個以上のVUを記録することができる領域を探す。また、マルチプレクサはPRUの多重化は行わない。その他のステップについては、上述の第1の実施例と同様であるため、その説明は省略する。

【0321】<アフレコ時の処理>次に、ユーザからアフレコが指示された場合の処理を、図19に沿って説明する。すでにアフレコの対象となるAV多重化ストリームに関するQuickTime管理情報はRAM102に読み込まれているものとする。

【0322】アフレコ開始位置のVUの先頭から再生用データの読み出しを行う(ステップ1101)。このとき、データを十分な再生時間分のデータを読み出すまで、ステップ1101を繰り返す(ステップ1102)。

【0323】ここで、十分な再生時間分のデータとは、再生用データ読み出しの中断期間が最大の場合でも再生が途切れないだけのデータ量を意味する。具体的には、PRUの記録(最悪約3秒)とAVデータの読み出しに伴う分断のジャンプ(最悪1秒)を連続的行った場合を想定し、4秒分のデータ量とする。

【0324】次に、ビデオデコード116とオーディオデコード115、およびオーディオエンコード118を起動する(ステップ1103)。オーディオエンコード118はサンプリングされた音声波形をAAUにエンコードし、AAUの周期でマルチプレクサに送る。6個のVUに相当する再生時間、つまり6秒分のオーディオデータのエンコードが終わったら、ホストCPUにエンコード終了を通知する。

【0325】また、ユーザからアフレコ終了を指示されていないかチェックする(ステップ1104)。指示されていないければ、エンコードが終了するまで、再生用AVデータの読み出しを行う(ステップ1106)。

【0326】エンコードが終了したら、オーディオストリームの記録位置にピックアップを移動し、アフレコデータを記録する(ステップ1107)。移動先は、アフレコ開始時には連続的な空き領域の先頭、それ以降は前回のアフレコデータ記録終了位置である。

【0327】尚、オーディオおよびビデオの最大ビットレートをそれぞれ256[kbps]、15[Mbps]としたとき、連続的に記録すべきオーディオデータの再生時間は<式18>より17.9秒となることから、連続的な空き領域のサイズは $17.9[\text{秒}] \times 256[\text{kbps}] = 572800[\text{byte}]$ 以上を確保する。アフレコデータ記録中に、連続した空き領域の終端に到達した場合、572800[byte]以上の連続した空き領域を探し、その先頭から記録を継続する。

【0328】アフレコ終了を指示されていれば、現在エンコード中のアフレコデータのエンコード完了を待つて

(ステップ1108)、そのオーディオストリームの終端へピックアップ107を移動し、アフレコデータを記録する(ステップ1109)。

【0329】最後に、上記オーディオストリームが上記AV多重化ストリームと同期して再生されることを示すように、上記AV多重化ストリームのQuickTime管理情報に上記オーディオストリームに対応するトラックを追加し、光ディスク106に記録する(ステップ1110)。

【0330】<再生時の処理>次に、ユーザから再生が指示された場合の処理を、図20に沿って説明する。すでに再生の対象となるAV多重化ストリームおよびオーディオストリームに関するQuickTime管理情報はRAM102に読み込まれているものとする。

【0331】まず、オーディオストリーム中の再生開始位置に対応する位置からオーディオデータを読み出す(ステップ1201)。読み出すデータ量は、 $2 \times N \times T_{av}$ に相当する再生時間とする。すなわち、ここでは12秒分のデータ量とする。また、変数 i に1をセットする。

【0332】再生開始位置のVUの先頭から再生用データの読み出しを行う(ステップ1202)。このとき、十分な再生時間分のデータを読み出すまで、ステップ1202を繰り返す(ステップ1203)。

【0333】ここで、十分な再生時間分のデータとは、再生用データ読み出しの中断期間が最大の場合でも再生が途切れないだけのデータ量を意味する。具体的には、アフレコデータの読み出し(最悪約4秒)とAVデータの読み出しに伴う分断のジャンプ(最悪1秒)を連続的行った場合を想定し、5秒分のデータ量とする。

【0334】次に、ビデオデコードおよびオーディオデコードを起動する(ステップ1204)。また、ユーザから再生終了を指示されていないかチェックする(ステップ1205)。指示されていないければ、再生開始から $N \times T_{av} \times i$ の時間が過ぎたかどうかチェックする(ステップ1206)。

【0335】過ぎていれば、 N 個のVUに相当する再生時間を持つアフレコデータの読み出しを行う。その後 i に1を加える。過ぎていなければ、ステップ1202と同様にVUの読み出しを行う(ステップ1207)。再生終了を指示されていれば終了する。

【0336】<第8実施例>次に、本発明における第8の実施例について、図21を用いて説明する。ここで、第8の実施例は上述した第1の実施例と類似するため、相違点に絞って説明する。尚、新たに定義していない記号は、第1の実施例における定義を用いる。

【0337】<AVストリームの形態>本実施例におけるAVストリームの構成について説明する。本実施例では、第1の実施例と異なり、VUの再生時間は1秒程度に限定しない。アフレコ対応ストリームは、図21に示すように、各VUの中にPRUを含む。

【0338】PRUには同じVU中のオーディオデータと同期再生されるオーディオデータが格納される。ここで

は、PRUをVUの先頭に配置しているが、オーディオデータとビデオデータとの間に置いても構わない。

【0339】尚、各VUはストリーム内でそれぞれ異なる再生時間であっても良いものとする。また、VU中(VU先頭も含む。ただし、AVストリームの先頭は除く)での分断は最大1回とする。一方、アフレコ非対応ストリームは、図21からPRUを除いた構成となる。

【0340】PRUの領域サイズの決定方法については、第1の実施例と共通であるため説明を省略する。

【0341】＜ディスク配置決定方法＞まず、アフレコ非対応ストリームにおけるVU再生時間の決定方法について説明する。ここでは、図7のリファレンス・デバイス・モデルにおいてシームレス再生が破綻しない、すなわちトラックバッファ502のアンダーフローがないようにVU再生時間を設定する。

【0342】トラックバッファ502がアンダーフローしないことを保証するには、最悪の条件においても、あるVUのデコード開始から次のVUのデコード開始までの間に少なくとも1個VUが読み出せれば良い。

【0343】ここで、最悪の条件について説明する。ま

$$T_{rmax} = T_{dmax} \times (R_v + R_a) / R_s = (T_{dmin} + d) \times (R_v + R_a) / R_s \quad \text{＜式35＞}$$

となる。

【0347】右辺第2項は、VU読み出し中の分断ジャンプにかかる時間を表す。ここでは、VU中の分断は最大1回であるため $1 \times T_a$ となる。＜式34＞に＜式35＞を代入し、 T_{dmin} でまとめると、

$$T_{dmin} \geq (d \times (R_v + R_a) + T_a R_s) / (R_s - R_v - R_a) \quad \text{＜式36＞}$$

が得られる。

【0348】つまり、データのビットレートに応じて、上記の式を満たすように、ストリーム中のVU再生時間の最大値 T_{dmax} と最小値 T_{dmin} とを決める必要がある。上式を満たしたとき、初期状態でトラックバッファ502にVUが1個存在すれば、その後ショック等の外乱が無い限り、常にシームレス再生が保証されることになる。

【0349】再生時間の決め方としては、まずVU再生時間最大値を決め、上式に基づき最小値を決めるという方法が考えられる。

【0350】トラックバッファ502のサイズは、少なくとも再生時間 T_{dmax} のVUをデコードしながら再生時間 T_{dmax} のVUを読み出しできるだけ用意する必要がある。トラックバッファ502中のVUのデコード済みの区間を再利用できない場合、そのサイズは $2 \times (R_v + 2R_a) \times T_{dmax}$ となる。

【0351】次に、アフレコ対応ストリームにおけるVUの再生時間の決定方法について説明する。上述した第6の実施例と同様、リファレンス・デバイス・モデルとリファレンス・アフレコ・アルゴリズムとを想定し、それらを用いてアフレコを行った際にシームレス再生が破綻しないように再生時間を設定する。

$$T_{rmax} = T_{dmax} \times (R_v + R_a + R_p) / R_s = (T_{dmin} + d) \times (R_v + R_a + R_p) / R_s \quad \text{＜式38＞}$$

ず、記号の定義を行う。ストリーム中のi番目のVUをVU#iとしたとき、VU#iの再生時間を $T_d(i)$ とする。また、ストリーム中の最小のVU再生時間を T_{dmin} 、最大のVU再生時間を T_{dmax} とする。尚、 $T_{dmax} = T_{dmin} + d$ の関係にあるとする。

【0344】このとき最悪の条件とは、 $T_d(i) = T_{dmin}$ 、 $T_d(i+1) = T_{dmax}$ であるとき、VU(i+1)の読み出し開始時にトラックバッファ502上にVU(i)しか存在しない場合である。なぜなら、読み出すべきVUが最も大きく、しかも読み出しに使える時間が最も小さいからである。

【0345】この条件においてVUの読み出しができれば、各VUのデコード直前には必ずVUがトラックバッファ502に存在するため、再生期間全体に関してもトラックバッファ502がアンダーフローすることはない。

【0346】前記最悪の条件においてVUを読み出せるためには、以下の式を満たす必要がある。

$$T_{dmin} \geq T_{rmax} + T_a \quad \text{＜式34＞}$$

ここで、右辺第1項の T_{rmax} は、再生時間 T_{dmax} のVUの読み出しにかかる時間を示し、

【0352】前記リファレンス・デバイス・モデルにおいて、前記リファレンス・アフレコ・アルゴリズムを用いてアフレコを行った場合、次のような条件を満たせば、アフレコ用バッファ504のオーバーフローおよびトラックバッファ502のアンダーフローがないことが保証できる。

【0353】その条件とは、最悪の条件においても、あるVUのデコード開始から次のVUのデコード開始までの間に少なくとも1個VUが読み出せることである。

【0354】ここで、最悪の条件とは、 $T_d(i) = T_{dmin}$ 、 $T_d(i-1) = T_d(i+1) = T_{dmax}$ であるとき、VU(i+1)の読み出し開始時にトラックバッファ502上にVU(i)しか存在しない場合である。なぜなら、読み出すべきVUが最も大きく、しかも読み出しに使える時間が最も小さいからである。

【0355】この最悪の条件において、VUの読み出しができれば、各VUのデコード直前には必ずVUがトラックバッファ502に存在するため、アフレコ期間全体に関してもトラックバッファ502がアンダーフローすることはない。

【0356】また、PRUエンコード完了に同期してアフレコデータのディスクへの記録を行っているため、アフレコ用バッファ504中のデータが累積していくことはなく、アフレコ用バッファ504のオーバーフローもない。

【0357】前述の最悪の条件において、VUを読み出せるためには、以下の式を満たす必要がある。

$$T_{dmin} \geq T_{rmax} + T_a + T_{wmax} \quad \text{＜式37＞}$$

ここで、＜式37＞右辺第1項の T_{rmax} は、再生時間 T_{dmax} のVUの読み出しに要する時間であり、

となる。

【0358】＜式37＞右辺第2項は、VU読み出し中の分断ジャンプにかかる時間を表す。ここでは、VU中の分断

$$T_{wmax}=3T_a+T_{dmax}\times R_p/R_s+T_y=2T_a+(T_{dmin}+d)\times R_p/R_s+T_y\cdots<式39>$$

となる。

【0359】ここで、 T_y はPRU両端が含まれるECCブロック中のアフレコデータ以外の最大記録時間を表しており、 $T_y=2\times 32KB/R_s$ となる。3回のアクセスはそれぞれPRUへの往復のアクセスおよびPRU記録中のアクセスを表

$$T_{dmin}\geq ((4T_a+T_y)\times R_s+d\times (R_v+R_a+R_p))/(R_s-R_v-R_a-2R_p)\cdots<式40>$$

となる。つまり、データのビットレートに応じて＜式40＞を満たすように、ストリーム中のVU再生時間の最大値 T_{dmax} と最小値 T_{dmin} とを決める必要がある。

【0361】尚、本実施例では、アフレコ対応ストリームとアフレコ非対応ストリームの双方において、VU中での分断回数を最大1回にしているが、任意の回数 N にしても構わない。このことによって、連続領域長を相対的に短くできるため、配置の自由度が高まる、という利点がある。その場合、＜式35＞右辺第2項の T_a に N を乗ずるように変更する必要がある。

【0362】また、本実施例においては、VU内(VU先頭を含む)での分断回数を最大1回にしているが、AVストリームを構成する各連続領域に必ず1回以上VUの先頭が含まれるというように制限しても良い。あるいは、各連続領域には必ず完全なVUが含まれるというように制限しても良い。

【0363】さらに、ストリーム内でVU再生時間が固定値 T_d の場合、連続領域長を(1)アフレコ対応ストリームの場合、 $T_d\times (R_v+R_a+R_p)$ 以上、(2)アフレコ非対応ストリームの場合、 $T_d\times (R_v+R_a)$ 以上に制限しても良い。

【0364】トラックバッファ502のサイズは、少なくとも再生時間 T_{dmax} のVUをデコードしながら再生時間 T_{dmax} のVUを読み出しできるだけ用意する必要がある。トラックバッファ502中のVUのデコード済みの区間を再利用できない場合、そのサイズは $2\times (R_v+R_a+R_p)\times T_{dmax}$ となる。

【0365】また、本実施例では、第1の実施例に説明した理由により、分断ジャンプと過去のRUへのピックアップの移動を非同期に行うことを想定しているが、分断ジャンプと過去のPRUへのピックアップの移動を同期して行うことを前提に T_{emin} を設定しても良い。この場合、＜式37＞の右辺第2項を取り除いて考えれば良い。

【0366】また、本実施例においては、リファレンス・アフレコ・アルゴリズムとしてPRUの含まれるECCブロックのみ記録しているが、第2の実施例のように、AVストリーム全体を再記録するようリファレンス・アフレコ・アルゴリズムを用いてもよい。その場合、＜式38＞の R_p を $(R_v+R_a+R_p)$ に置き換えることになる。

【0367】本実施例と第6の実施例との違いは、本実施例においてVUデコード開始時にトラックバッファ502

は最大1回であるため $1\times T_a$ となる。また、＜式37＞右辺第3項の T_{wmax} は、再生時間 T_{dmax} のPRUの記録に要する時間であり、

す。尚、第6の実施例と同様、PRUを常に連続領域に記録するようにすれば、アクセスは2回で済む。

【0360】＜式37＞に＜式38＞、＜式39＞を代入し、 T_{dmin} でまとめると、

中にVUが少なくとも1個あれば、ショック等の外乱がなければその後のアフレコしながらのシームレス再生を保証できるのに対し、第6の実施例は保証できない点にある。記録、再生、アフレコ処理に関しては、上述の第6の実施例と同様である。

【0368】

【発明の効果】以上説明したように、本発明によれば、AVストリーム中を連続的に記録する単位を、ピックアップ移動性能、データ転送レート、データのビットレート、アフレコ領域のデータ書き換えの制御、のうちのーによって決定することで、リアルタイムアフレコを確実に行うことが可能になる。

【図面の簡単な説明】

【図1】本発明の実施形態における概略構成を示すブロック図である。

【図2】QuickTimeファイルフォーマットにおける管理情報とAVストリームとの関係を示す説明図である。

【図3】QuickTimeファイルフォーマットにおける管理情報の概要を示す説明図である。

【図4】本発明の第1の実施例におけるストリームの構成を示す説明図である。

【図5】本発明の第1の実施例におけるアフレコ非対応VUの構造を示す説明図である。

【図6】本発明の第1の実施例におけるアフレコ対応VUの構造を示す説明図である。

【図7】本発明の第1の実施例におけるリファレンス・デバイス・モデルを示す説明図である。

【図8】本発明の第1の実施例におけるリファレンスアフレコ・アルゴリズムを示す説明図である。

【図9】本発明の第1の実施例における記録動作を示すフローチャートである。

【図10】本発明の第1の実施例におけるアフレコ動作を示すフローチャートである。

【図11】本発明の第1の実施例における再生動作を示すフローチャートである。

【図12】本発明の第2の実施例におけるリファレンスアフレコ・アルゴリズムを示す説明図である。

【図13】本発明の第4の実施例における第1のアフレコ動作を示すフローチャートである。

【図14】本発明の第4の実施例における第2のアフレ

コ動作を示すフローチャートである。

【図15】本発明の第5の実施例におけるアフレコ動作を示すフローチャートである。

【図16】本発明の第6の実施例におけるアフレコ対応ストリームの構成を示す説明図である。

【図17】本発明の第6の実施例におけるリファレンスアフレコ・アルゴリズムを示す説明図である。

【図18】本発明の第7の実施例におけるリファレンスプレイバック・アルゴリズムを示す説明図である。

【図19】本発明の第7の実施例におけるアフレコ動作を示すフローチャートである。

【図20】本発明の第7の実施例における再生動作を示すフローチャートである。

【図21】本発明の第8の実施例におけるアフレコ対応ストリームの構成を示す説明図である。

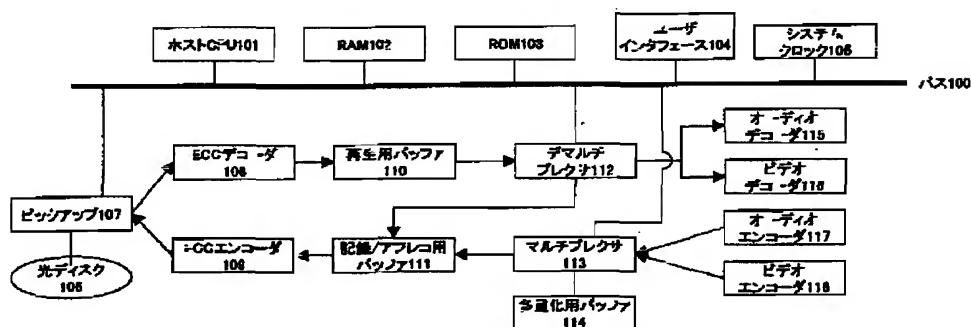
【図22】従来技術におけるディスク上での記録形態を示す説明図である。

【図23】従来技術におけるアフレコ時のヘッドの動きとバッファメモリ108におけるデータの占有率の変化を示す模式図である。

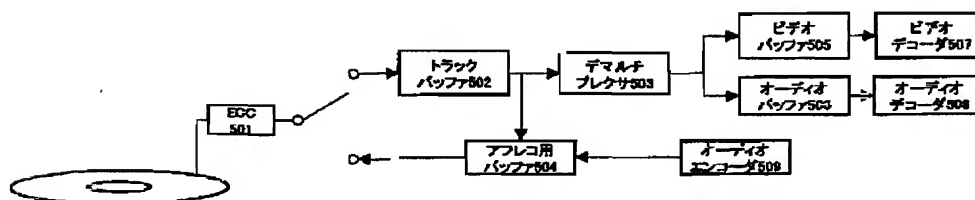
【符号の説明】

- 100 バス
- 101 ホストCPU
- 102 RAM
- 103 ROM
- 104 ユーザインタフェース
- 105 システムクロック
- 106 光ディスク
- 107 ピックアップ
- 108 ECCデコーダ
- 109 ECCエンコーダ
- 110 再生用バッファ
- 111 記録/アフレコ用バッファ
- 112 デマルチプレクサ
- 113 マルチプレクサ
- 114 多重化用バッファ
- 115 オーディオデコーダ
- 116 ビデオデコーダ
- 117 オーディオエンコーダ
- 118 ビデオエンコーダ

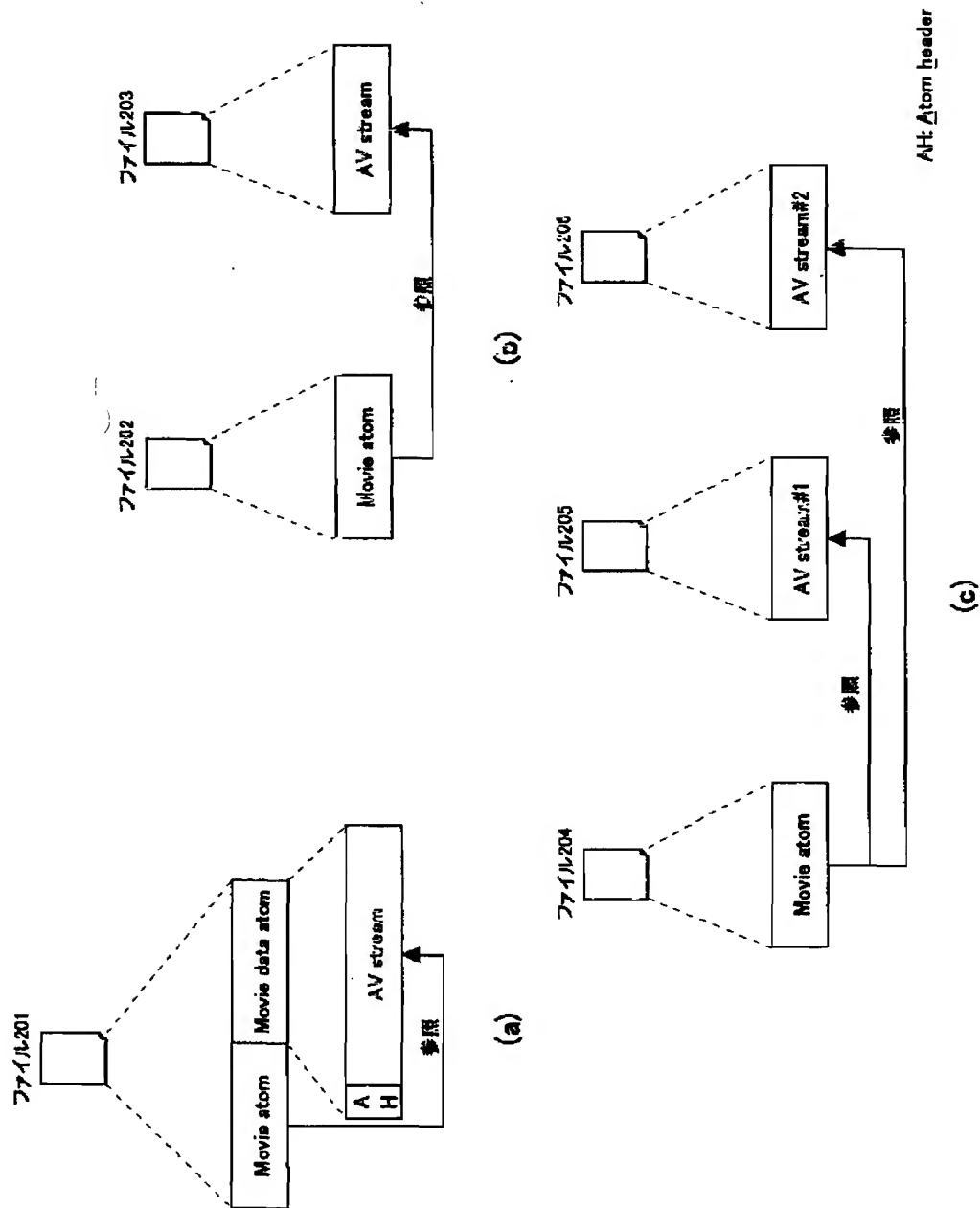
【図1】



【図7】



【図2】



【図3】

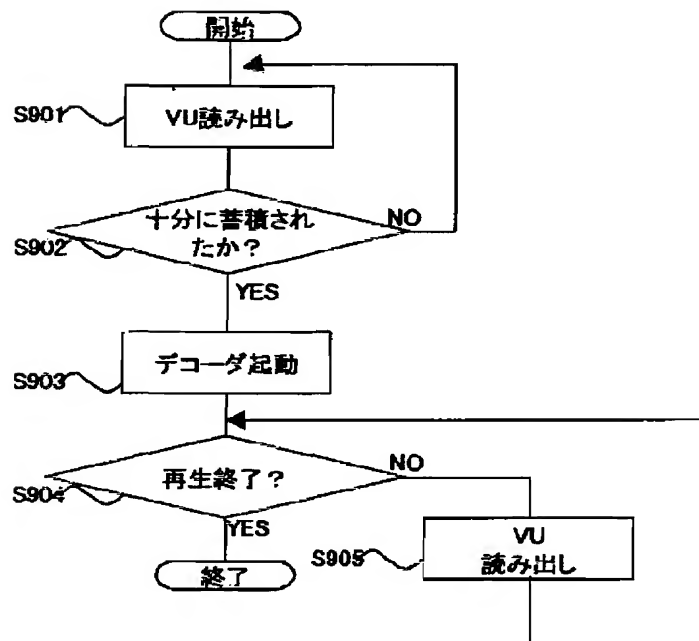
```
Movie atom {  
    movie header atom  
    track atom (video track)  
    track atom (main audio track)  
    :  
}
```

(a)

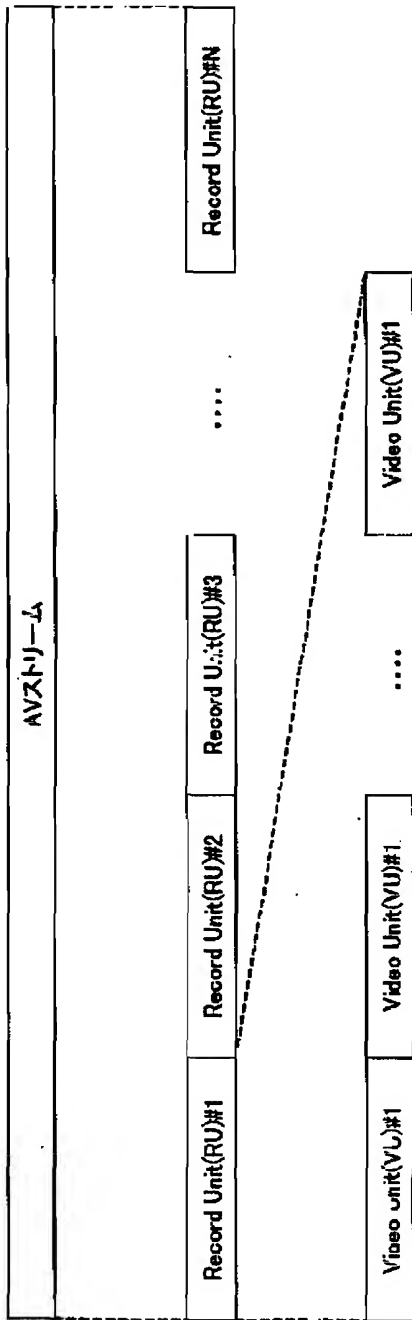
```
X descriptor atom {  
    record-unit description atom  
    sst performance atom  
}
```

(b)

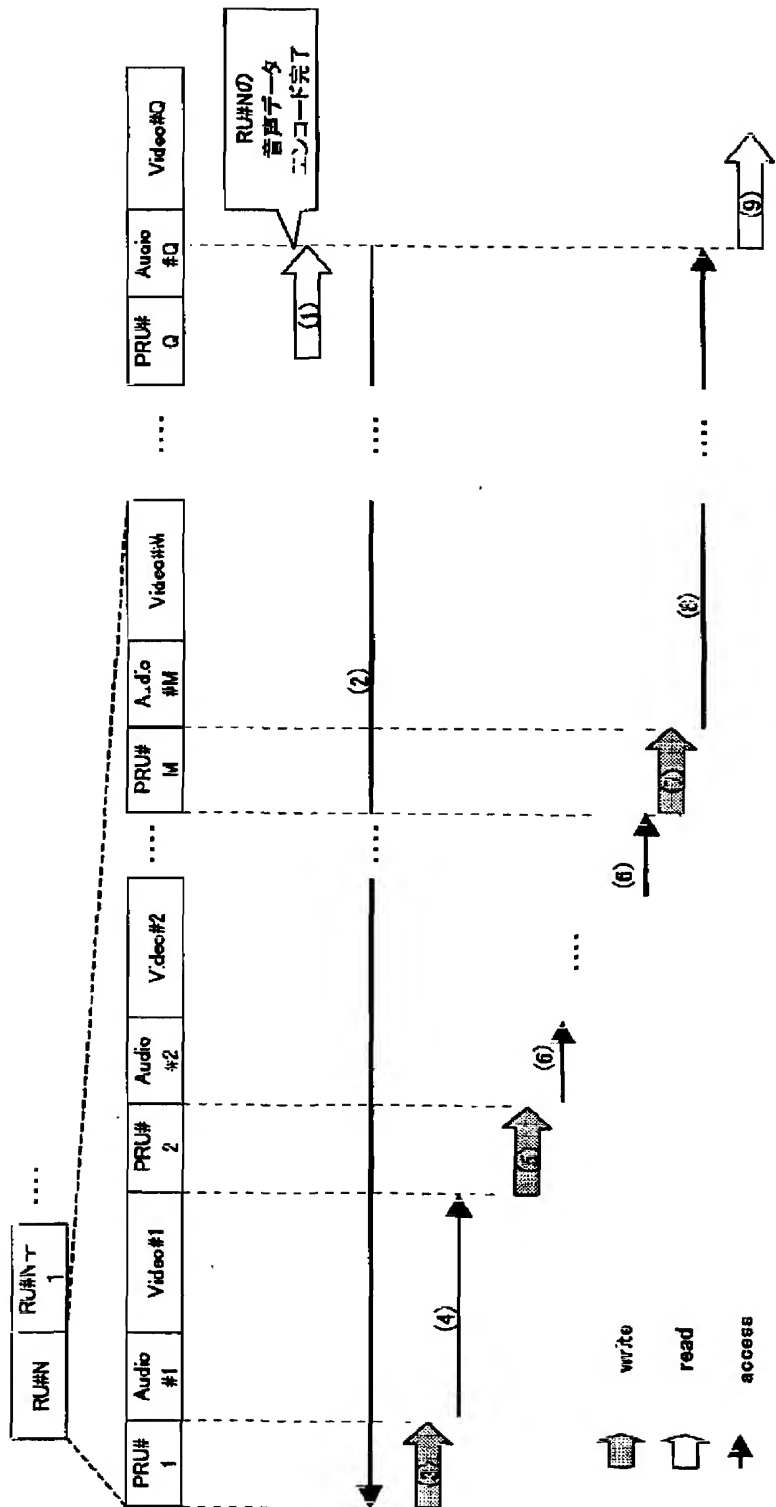
【図11】



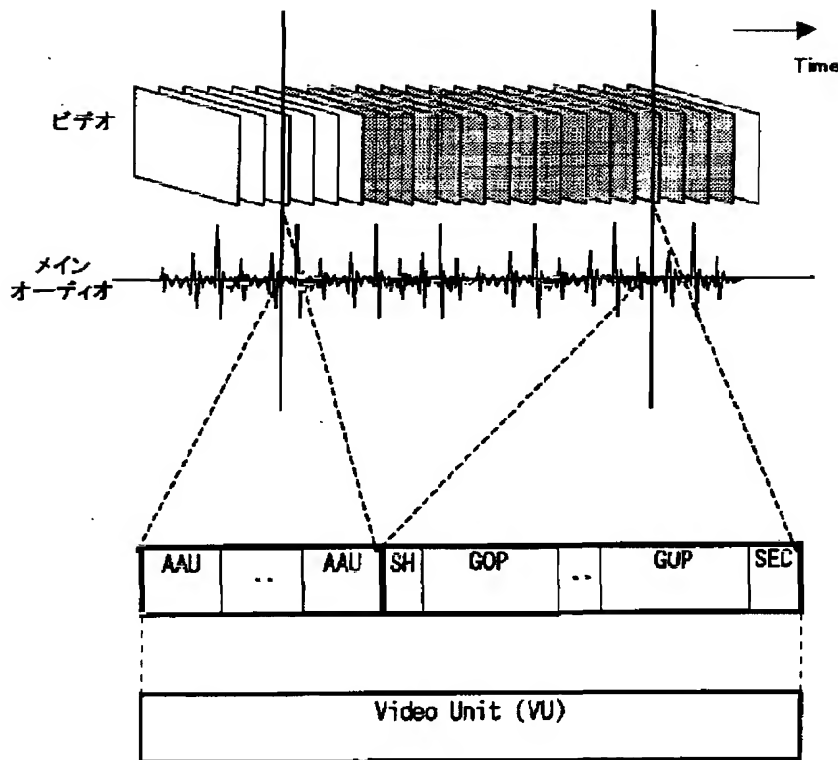
【図4】



【図8】

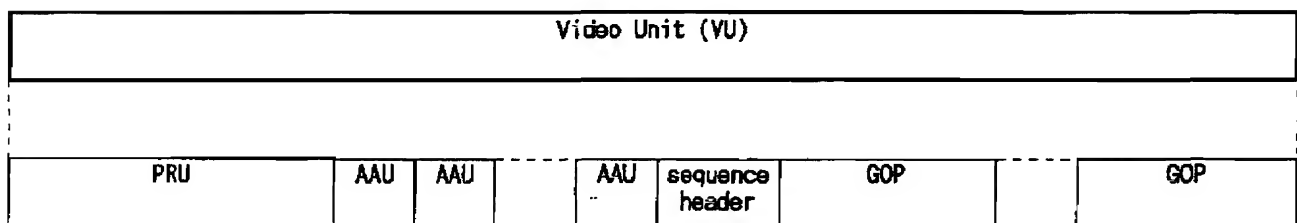


【図5】

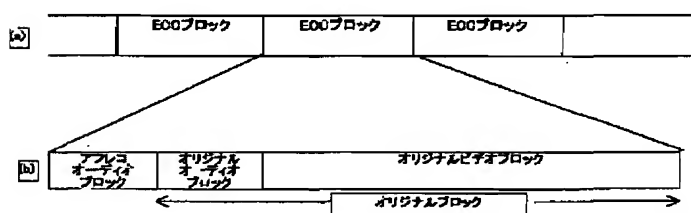


SH: Sequence Header
SEC: Sequence End Code

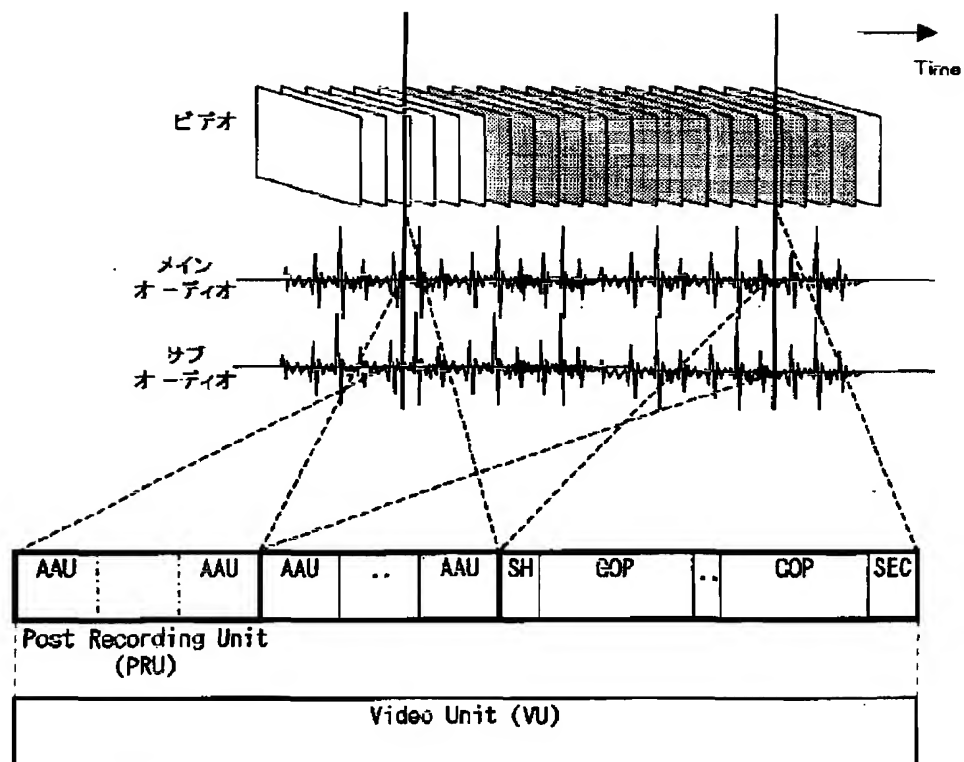
【図21】



【図22】

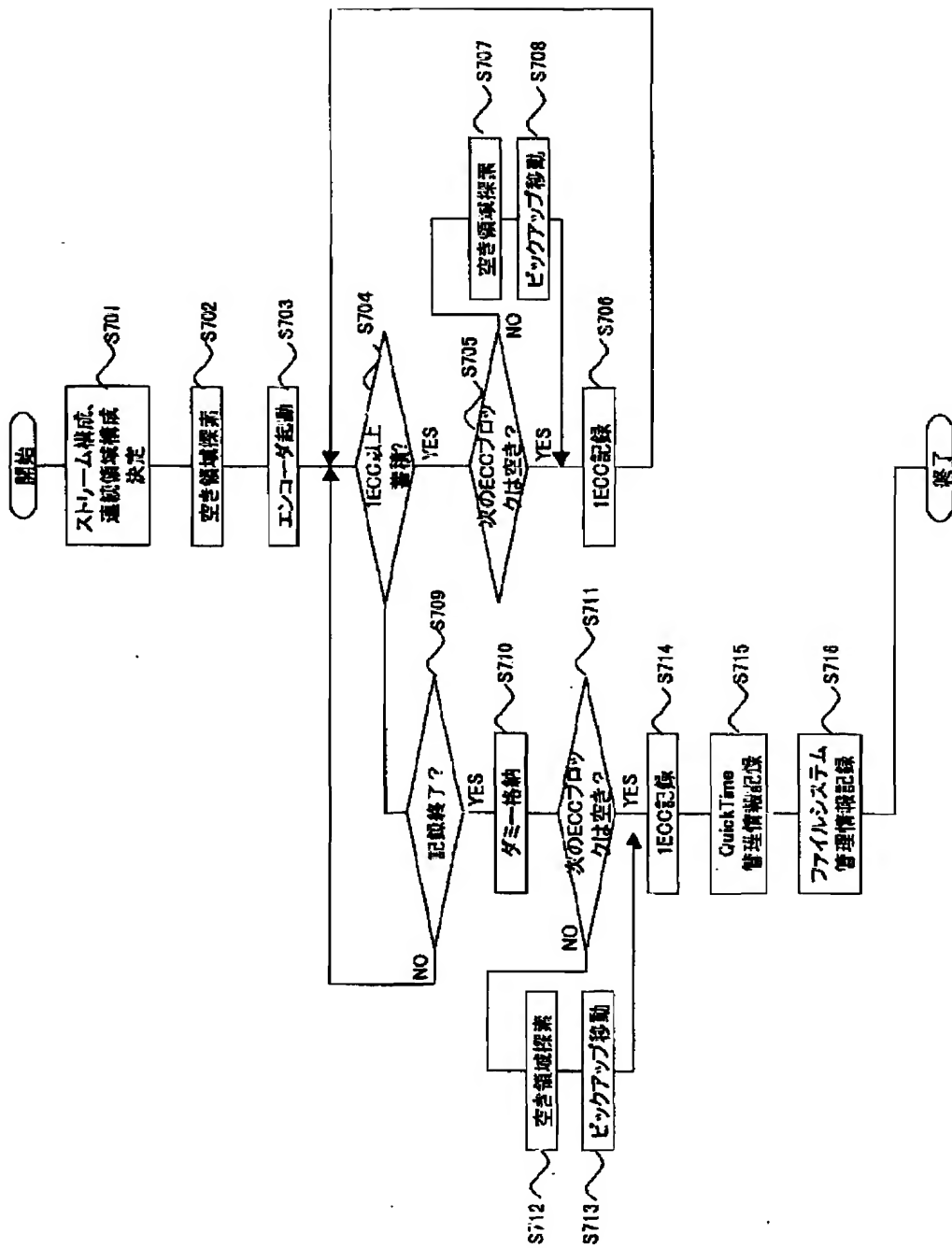


【図6】

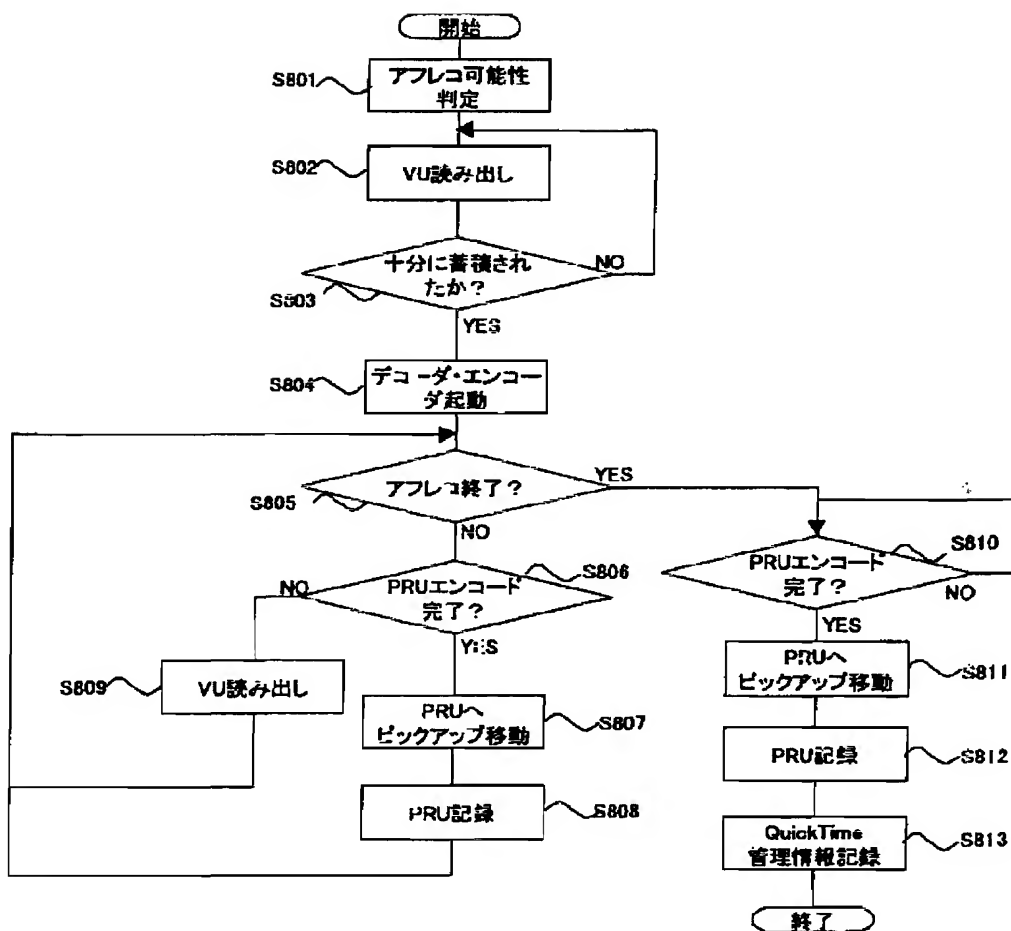


SH: Sequence Header
SIC: Sequence Ind Code

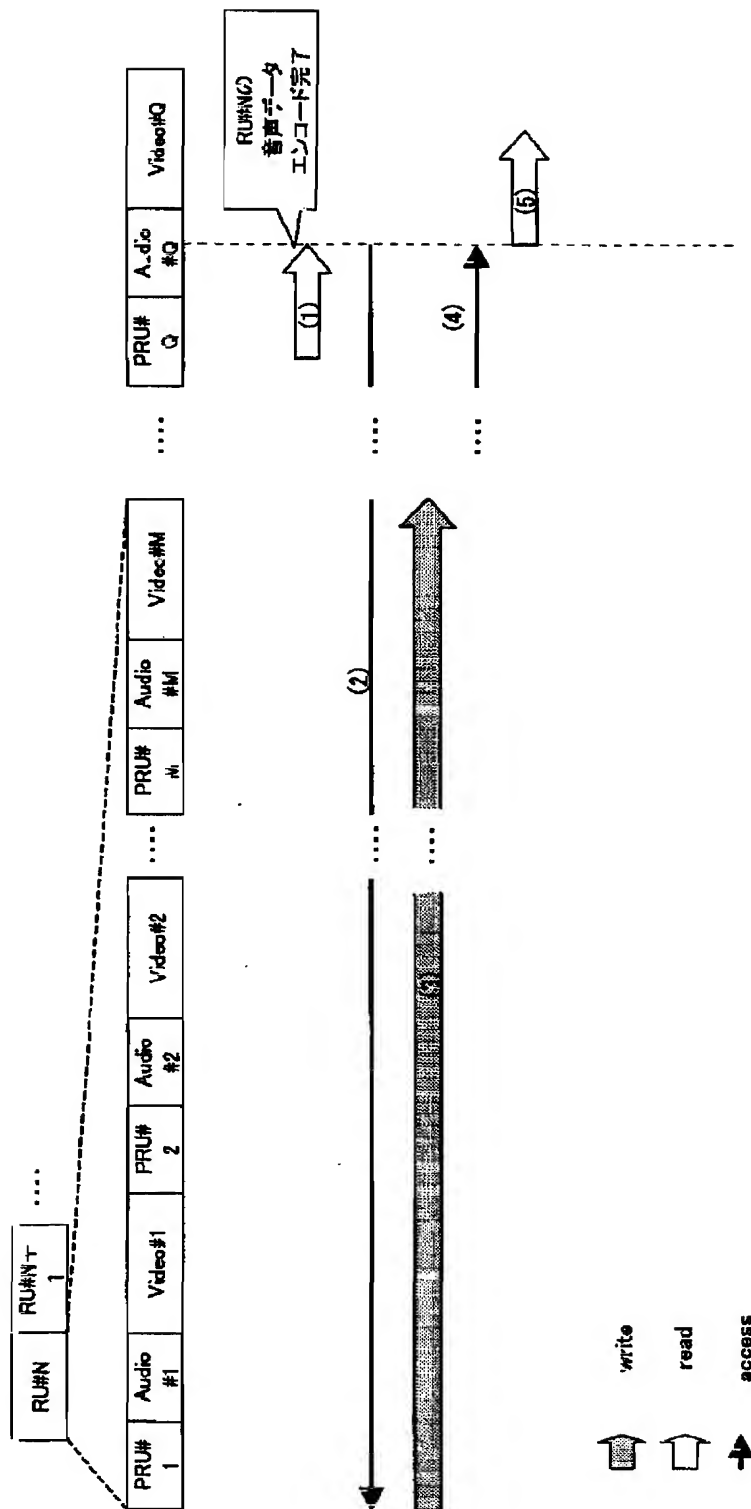
【図9】



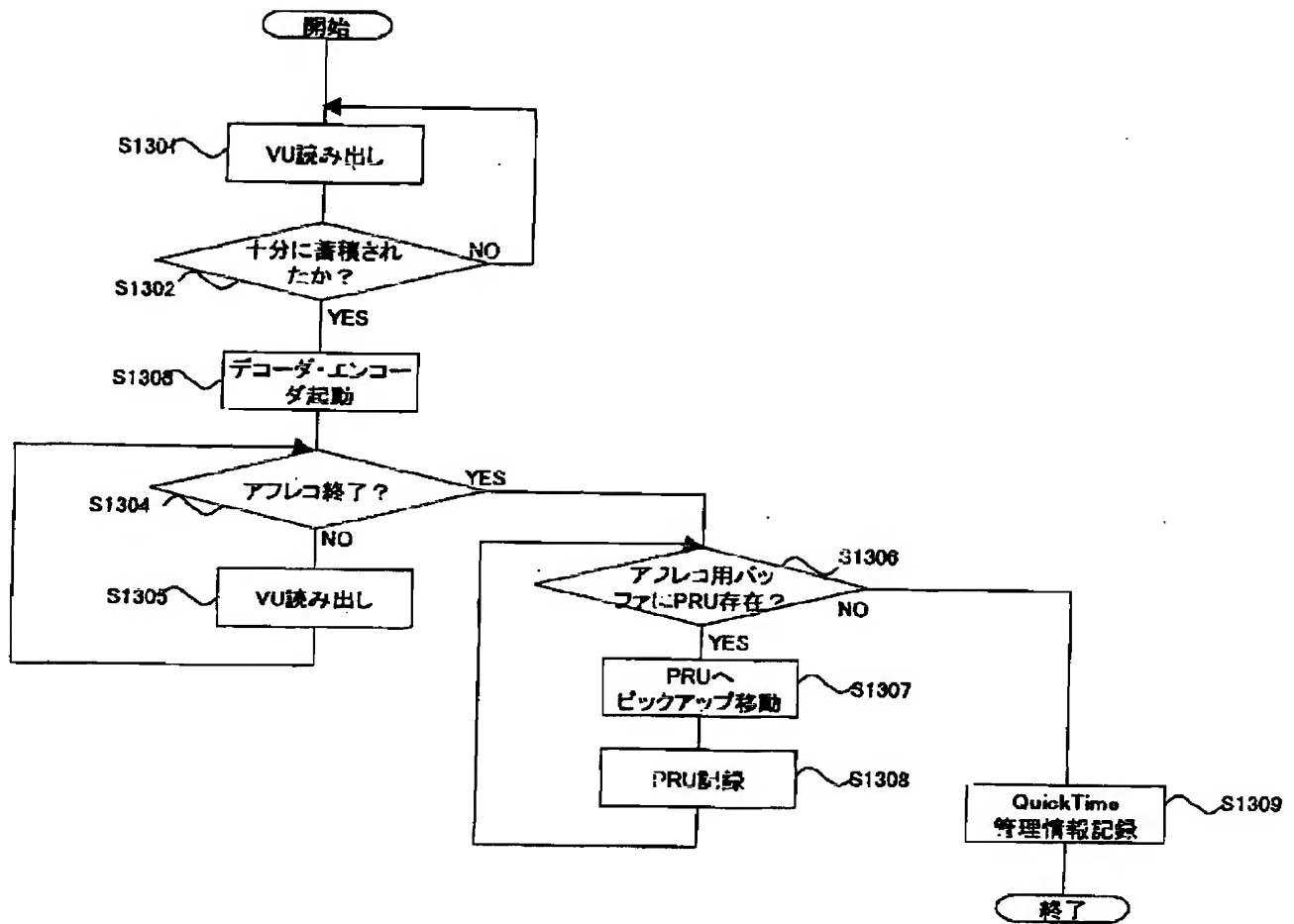
【図10】



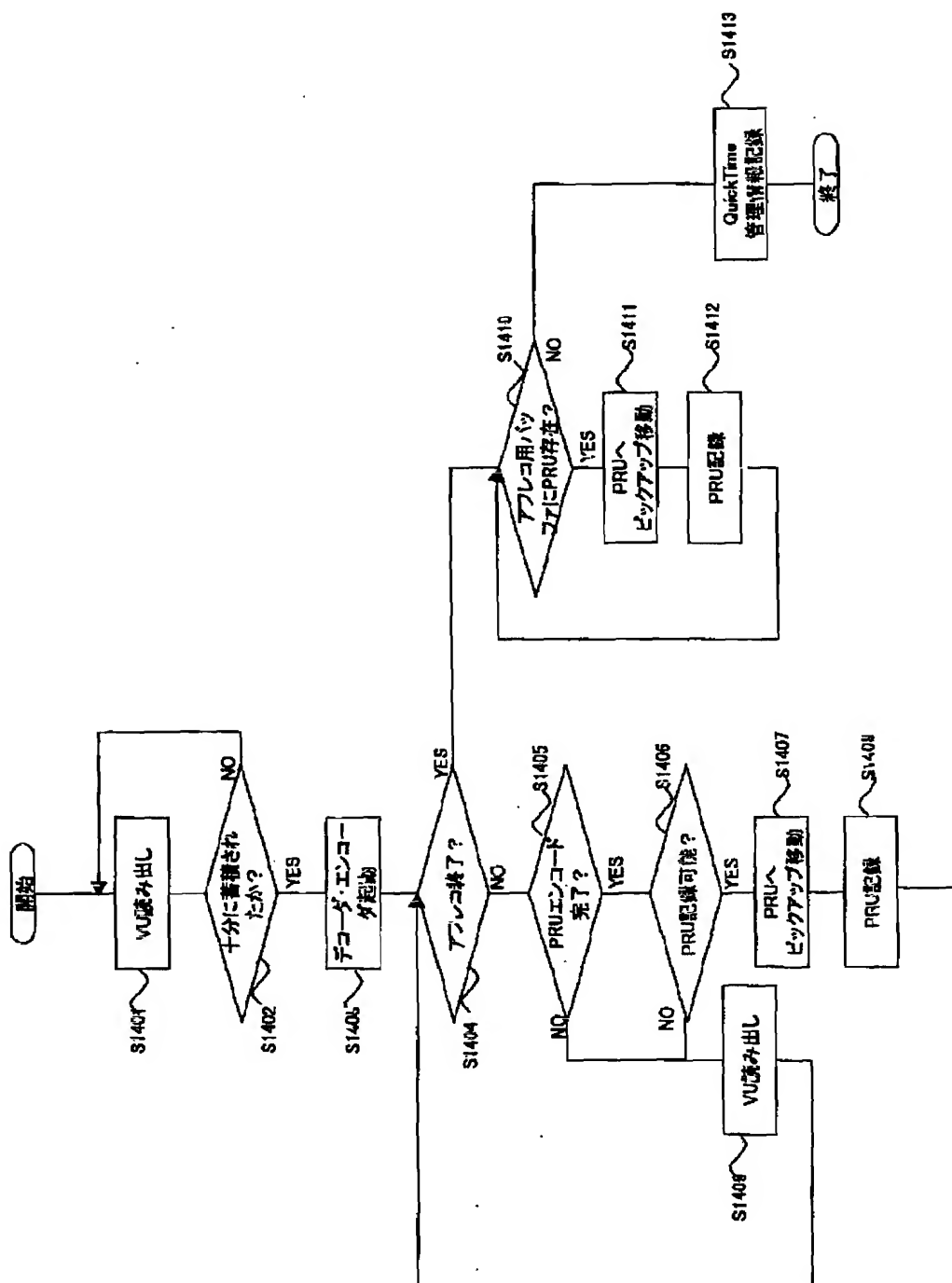
【図12】



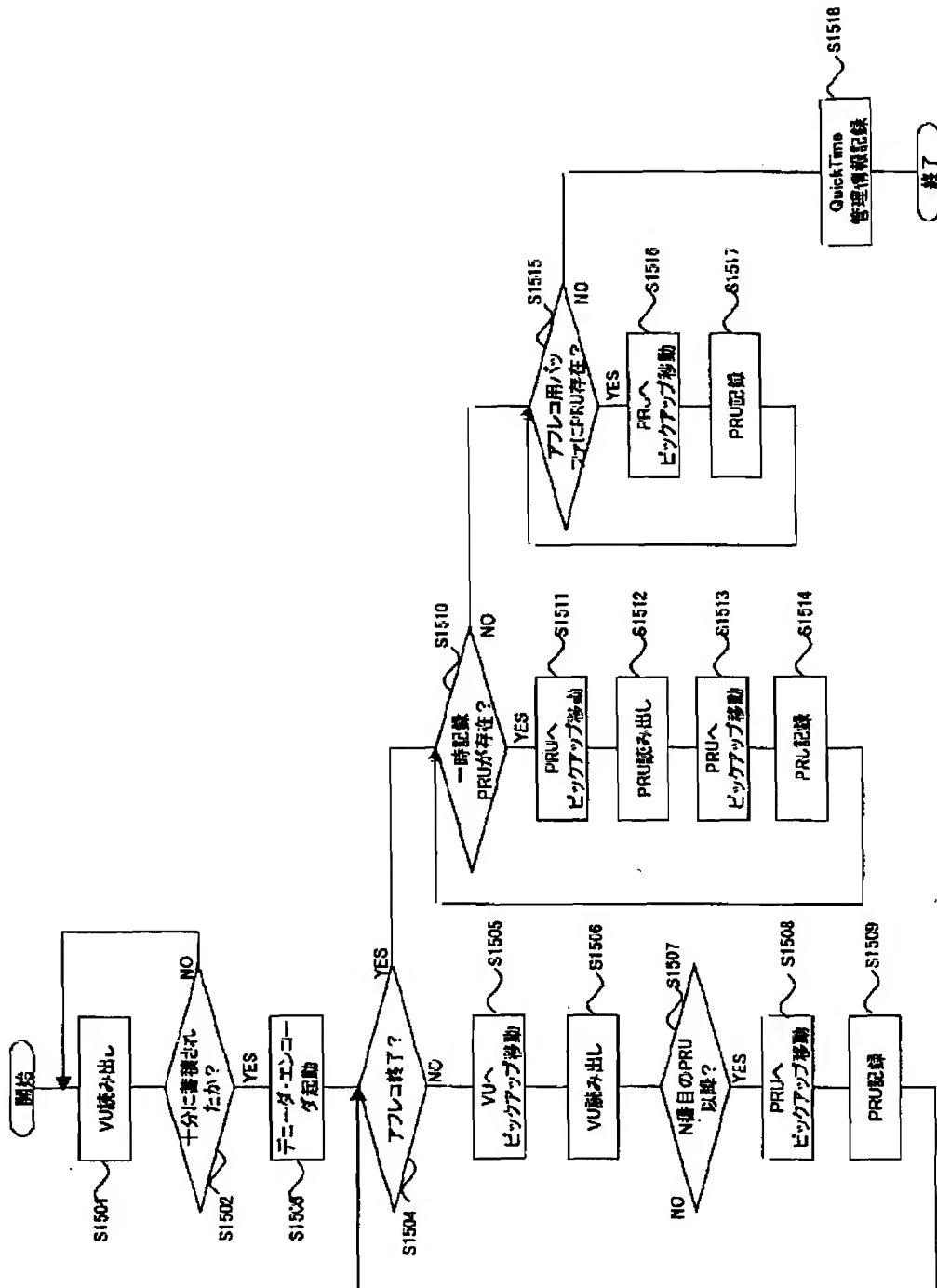
【図13】



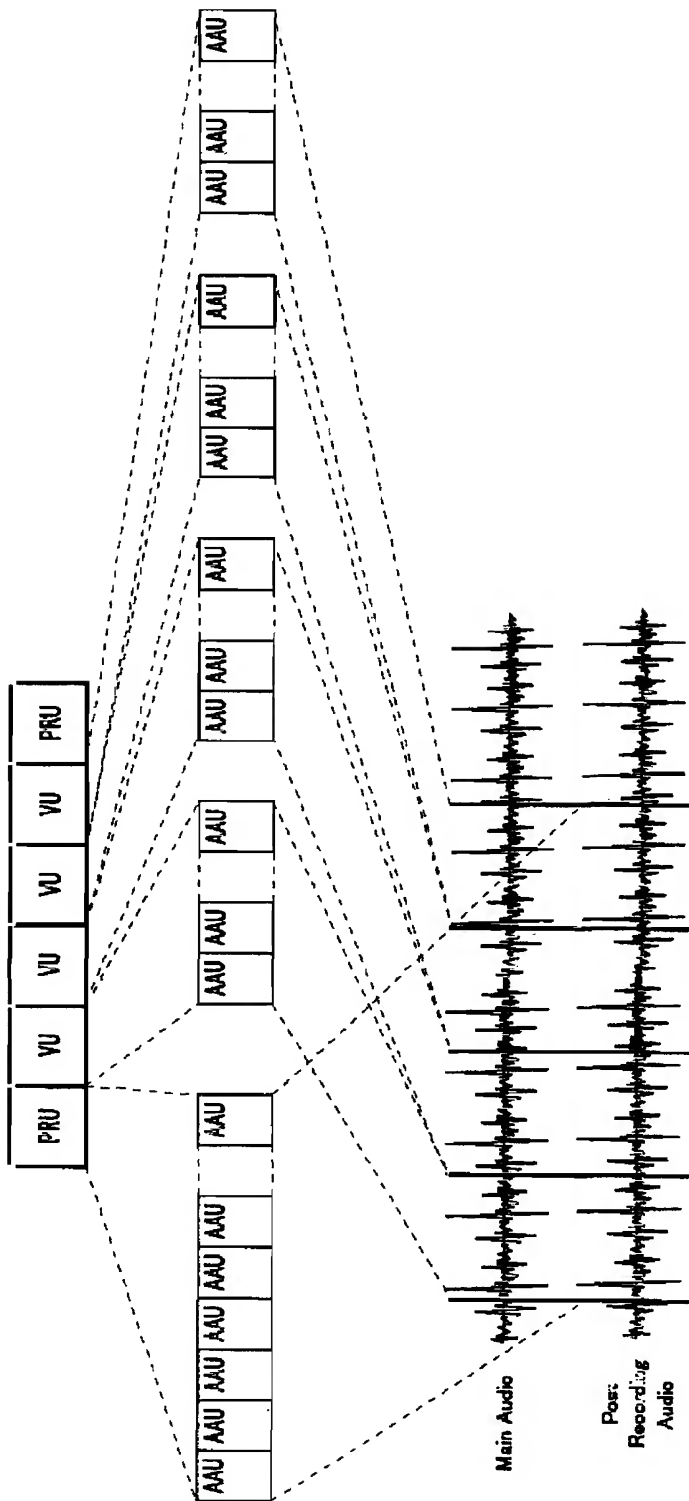
【図14】



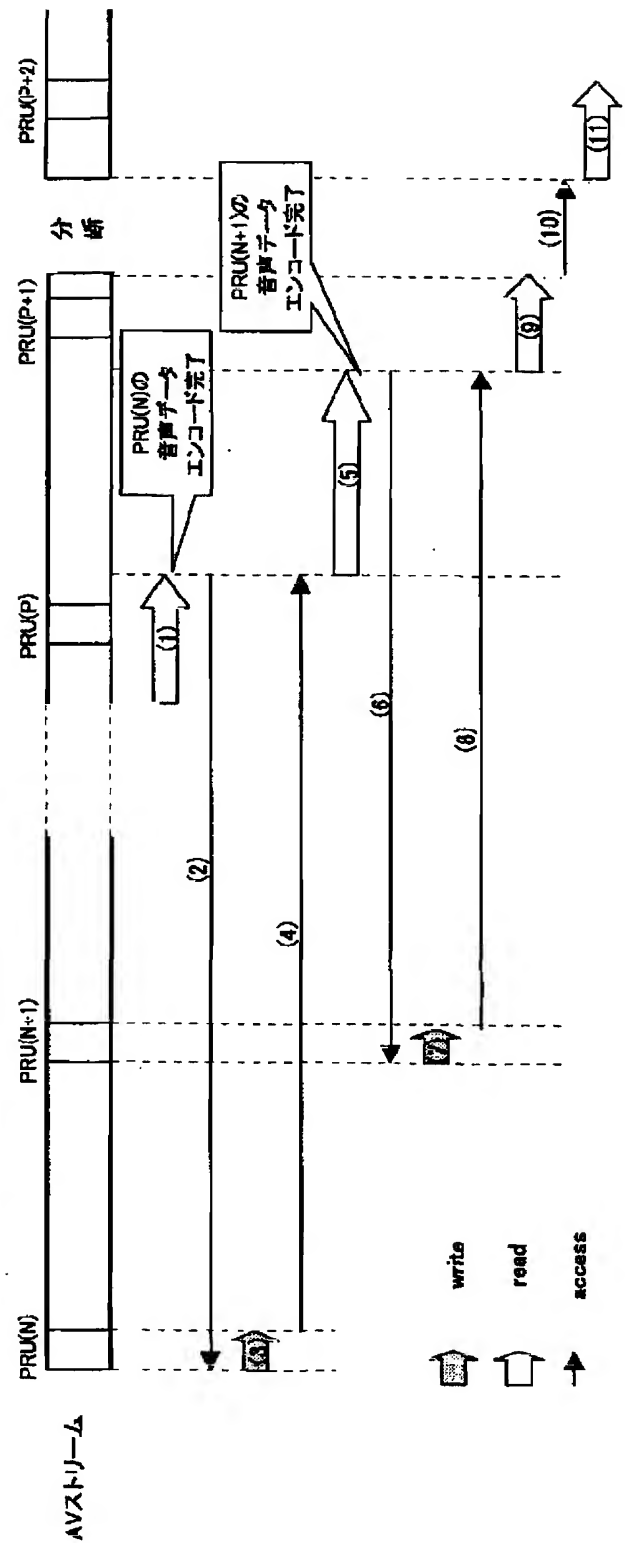
【図15】



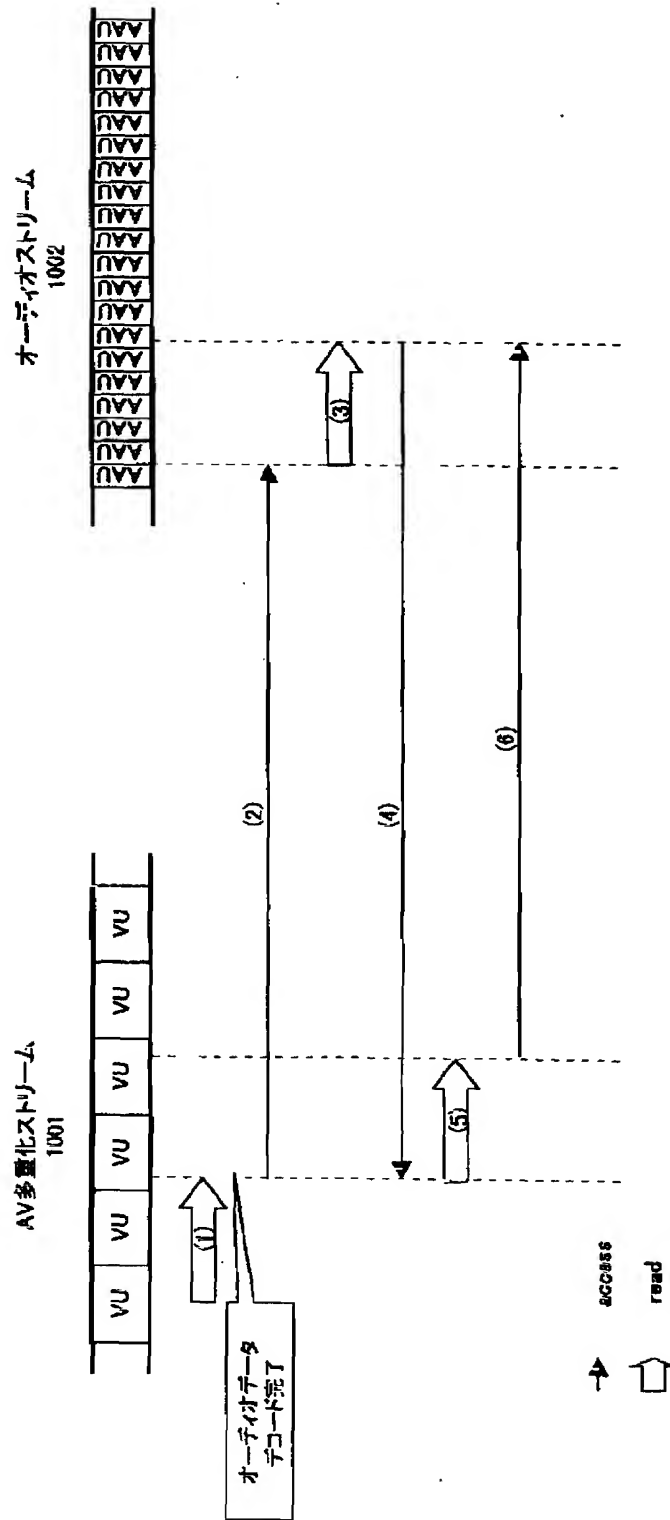
【図16】



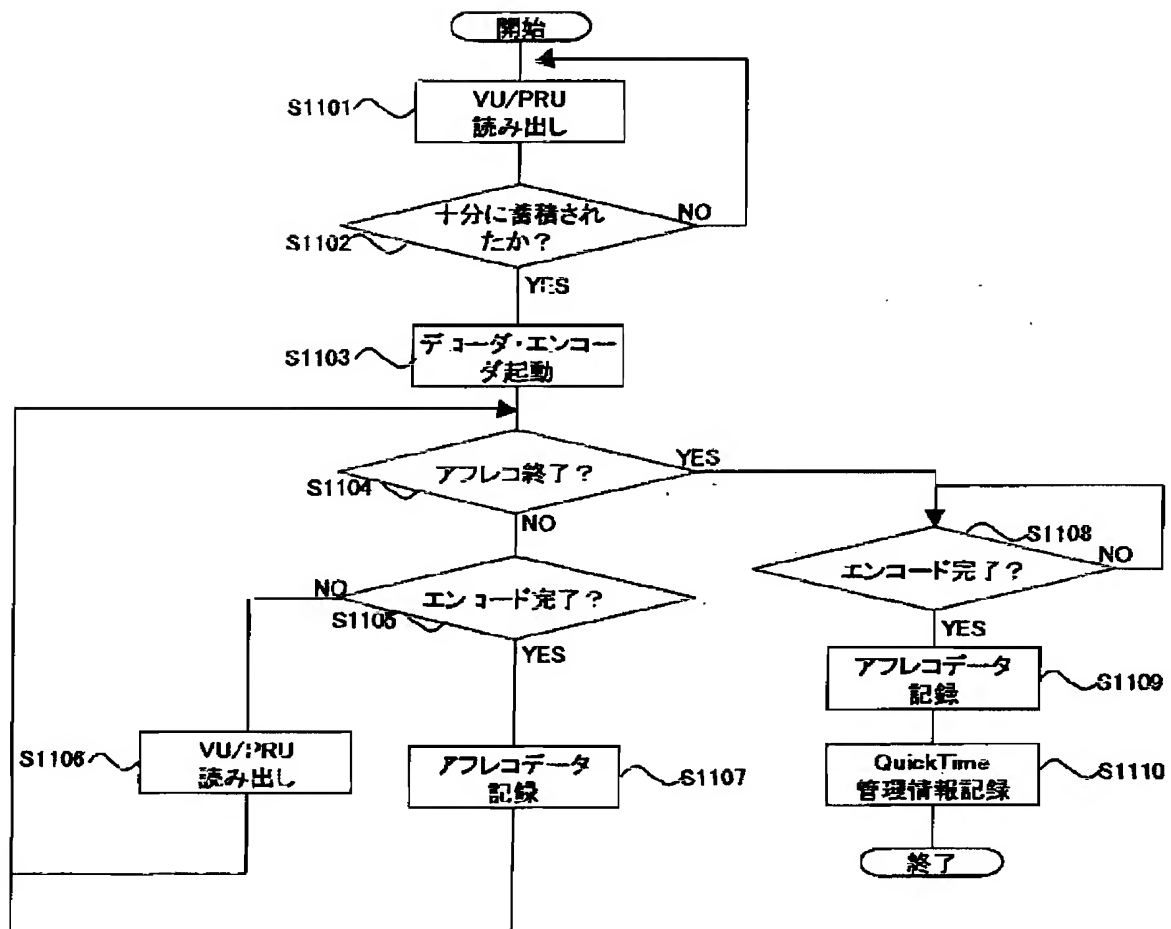
【図17】



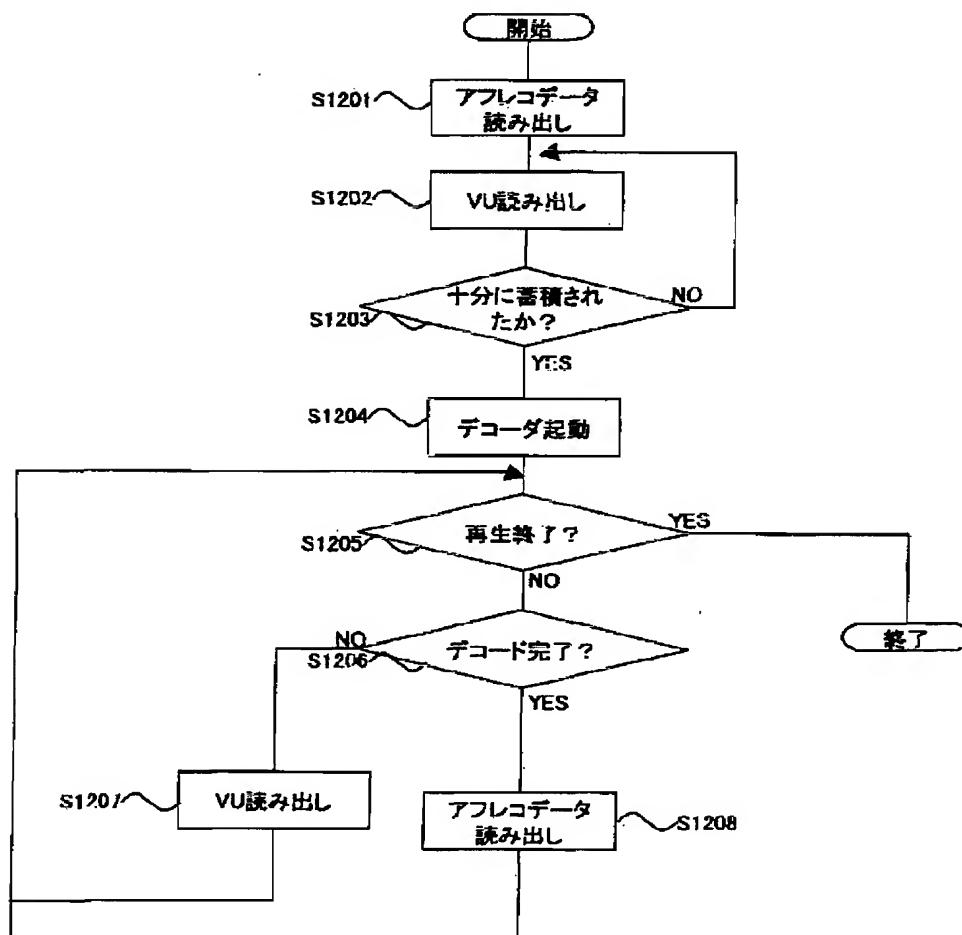
【図18】



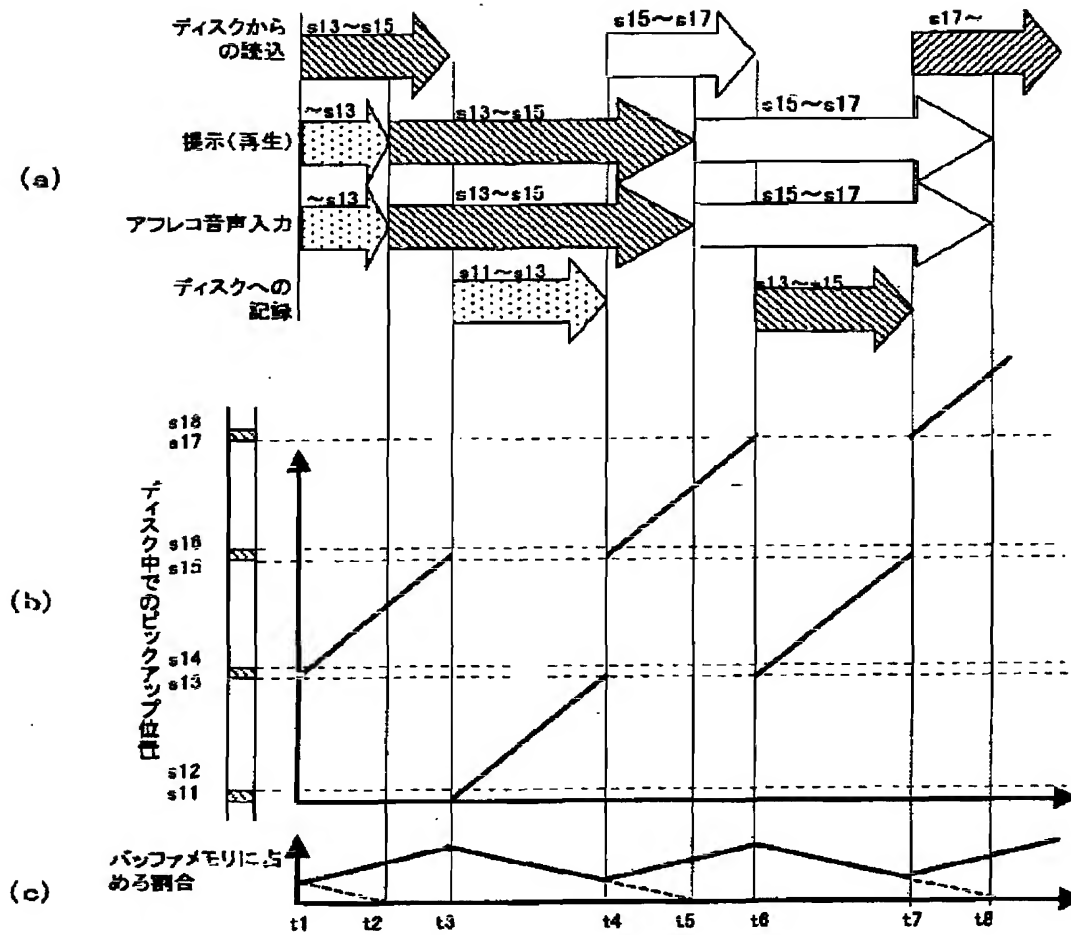
【図19】



【図20】



【図23】



フロントページの続き

(51) Int. Cl.⁷

G 1 1 B 27/034

H 0 4 N 5/91

識別記号

F I

H 0 4 N 5/91

(参考)

G 1 1 B 27/02

C
N
Z
K

(72) 発明者 山口 孝好

大阪府大阪市阿倍野区長池町22番22号 シ

ャープ株式会社内

F ターム(参考) 5C053 FA14 FA23 GB01 GB05 GB11
JA01 JA03 JA05 LA11
5D044 AB05 AB07 AB10 BC06 CC06
DE02 DE03 DE12 DE14 DE48
DE54 DE92 EF03 EF05 FG23
GK08 GK12
5D090 AA01 BB04 CC01 CC04 CC14
DD03
5D110 AA17 AA27 AA29 CA07 CF05
DB02